0.1 Modelling the ITC2021 Problem

Phased or unphased!

"In this competition we only consider time-constrained double round-robin tournaments with even numbers of teams" (each team plays exactly one game per timeslot)

time-constrained/compact: uses the minimal number of time slots needed otherwise: time-relaxed

timeslots: single timeslot with parallel assignments for multiple games tasks: team vs. team game

2RR: - if (a, b) played in the first half, then (b, a) must play in the second round

create a group for each team create a task for each team x team game add them to the corresponding

0.1.1 Phased

In **phased** competitions, the competition is split into intervals in which all combinations of teams compete once.

conjunction of group requirements

one per phase problem: determining phase length

0.1.2 Capacity Constraints

CA1: teams, max, mode: H for home, A for away, slots, type: HARD or SOFT; teams from cannot play more than max games of mode in timeslots slots (in benchmarks teams only ever contains one team - other benchmarks can be expanded into that format)

CA2: teams1, min, max, mode1: H for home, A for away, HA for both, mode2: GLOBAL ???, teams2: played against teams, slots, type

CA3: teams1, max, mode1, teams2, intp, mode2, type;

0.1.3 Notes reading

- MetaData can be ignored - here: only single league with all teams (encoding should accomodate multiple though) - here: no grouping teams and time slots into time groups (could be achieved easily with groups?) - RobinX is general format? not from ITC? -

1. create timeslots (only ID and name) - calculate max possible number of games per timeslot and create that many instances for each (problem: many symmetric assignments?)
2. create a group for each team
3. (league: id, name)

4. (need group for each league for rules?)
5. (team: id, league, name)
6. create group for each team
7. read structure: for each league:
   (a) create tasks for each team x team combination in a league, depending on round-robin number, add team groups
   (b) set compactness rule (???) if applicable
   (c) set game mode (phased/unphased) rule (???) if applicable
8. The competition instances only use 9 of the possible constraints (these should be implemented here)

0.1.4 Components

Slots: group: each timeslot has a group with its id
Games: name: IDteam1_IDteam2 groups:

1. group: "H" or "A" for home or away for each time
2. group "HA" for both teams
3. "l" league
4. groups with the id of each team

In total 8 groups per game.
Assignment: One fixed timeslot, $\lfloor \#games/2 \rfloor$ optional game slots
Implicit Rules: // a team can only play one game simultaneously

```
for( assignment ){
        itc21.addRule( Unique(gameType) );
```

rules used:

- AssignOnce(component)
- Unique(assignment, slotType)
-

Rules:
CA1: "teams t each play at most max games of mode m in slots s", can be hard or soft "each team in temas triggers a deviation equal to the number of mode games in slots more than max"
components: groups (team), timeslots (), for each team (group): NumAssigned(InComponent(Timeslots ts...) & InGroup(Mode, mode)) ¡= MAX

CA2: "each team in teams1 does not play more than max games of mode m against teams2 during timeslots slots" for each team in

NumAssigned(InTimeslots(slots) & Groups(teams2) & Group(mode)) ¡= MAX

CA3: "Each team in teams1 plays at most max games of mode m against teams2 " For each group in teams1: ???? INTP ???? NumAssigned()

C4: "teams1 play at most max games of mode m against teams2 during time slots slots"

NumAssigned(OR(Group(teams1)) & OR(Group(teams2)) & Group(mode) & In(slots)) ¡= MAX

GA1: (min max meetings slots type) "fixed and forbidden time slot assignments" "at least min and at most max games from meetings = $\{(i_1, j_1), (i_2, j_2), ...\}$ take place in time slots slots"

(if min = max add single equality constraint)

NumAssigned() ¿= min NumAssigned() ¡= max

(Break constraints)

"If a team plays a game with the same home-away status as its previous game, it is called a break"

BR1: (teams intp mode2 slots type) "Each team in teams has at most intp home/away breaks (H,A,HA) during time slots slots"

"BR1 constraints with multiple teams can be broken up into one constrant per teams, therefore ITC2021 only has single team constraints"

NumAssigned(InComponent(Timeslot, slots) & 'prevGame.mode = curGame.mode') ¡= INTP

BR2