

Simulazione AMPL del calendario sportivo di serie A

PROGETTO DI RICERCA OPERATIVA

MICHELE PORTA

MATRICOLA: 289672

Il progetto

Realizzare un modello matematico per definire le partite di andata e ritorno di un campionato di calcio tenendo in considerazione i seguenti vincoli:

- Limitare al massimo a 2 partite consecutive in casa o in trasferta per ogni singola squadra
- Coppie di squadre che condividono lo stesso impianto sportivo non potranno giocare in casa contemporaneamente
- A causa di eventi concomitanti, certe squadre non potranno giocare in casa in certe giornate prefissate

L'obiettivo è quello di minimizzare il numero di partite consecutive giocate in casa o in trasferta di ogni squadra.

Si tratta di un problema di programmazione lineare.

Modello matematico

Insiemi

Sono stati definiti i seguenti insiemi:

- **Squadre**: l'insieme che conterrà l'elenco delle squadre partecipanti al campionato.
- **Stadio**: l'insieme che conterrà tutti gli stadi utilizzabili per gli incontri del campionato
- **Incontro**: insieme definito come 3-tupla {Squadre, Squadre, Stadio} per rappresentare l'incontro sportivo e l'impianto nel quale si svolgerà
- **Giornate**: l'insieme numerico che conterrà il numero di giornate dell'intero campionato
- **Eventi**: insieme definito come tupla {Giornate, Stadio} nel quale verranno elencati gli eventi concomitanti che limiteranno l'utilizzo degli impianti in certe giornate (ad esempio causa manutenzione, concerto ecc..)
- **Campo**: insieme definito come tupla {Squadre, Stadio} nel quale viene specificato per ogni squadra quale è il proprio stadio di appartenenza.

Parametri

Sono stati definiti i seguenti parametri:

- **Num_squadre**: parametro da settare con il numero di squadre che disputeranno il campionato
- **Peso**: parametro che determinerà il peso di una partita rispetto ad un'altra per una determinata giornata. In questo caso verrà settata a 1 in ogni incontro perché inizialmente nessun incontro ha la priorità rispetto ad un altro.

Variabili

È stata definita la seguente singola variabile:

- **X**: rappresenta una variabile binaria composta dalla tupla {Incontro, Giornata}. Conterrà quindi 4 elementi (squadra 1, squadra 2, stadio squadra 1 e numero giornata). La variabile binaria restituirà 0 o 1 a seconda dei casi.

Vincoli

Sono stati definiti 9 vincoli nominati S_i con $i \in \{1..9\}$ ognuno con la propria funzione:

- **S1**: Massimo (numero di squadre/2) incontri per ogni giornata
- **S2**: Una squadra può giocare una sola volta a giornata
- **S3**: Ogni incontro è unico e deve svolgersi in una sola giornata
- **S4**: Uno stadio può essere occupato al massimo una volta a giornata
- **S5**: Per ogni evento stabilito inizialmente in uno stadio alla i -esima giornata non ci può essere un incontro disputato nello stesso stadio alla stessa i -esima giornata
- **S6**: Una squadra può giocare in casa consecutivamente al massimo 2 volte consecutivamente
- **S7**: Una squadra può giocare fuori casa consecutivamente al massimo 2 volte consecutivamente
- **S8**: La partita giocata all'andata deve ripetersi a campo invertito nel girone di ritorno
- **S9**: La partita giocata al ritorno deve essere stata giocata a campo invertito nel girone di andata

Obiettivo

È stato definito un solo obiettivo da minimizzare.

- **Partite_consecutive**: minimizzare il numero di partite consecutive giocate in casa o in trasferta per ogni squadra.

Trasformazione in AMPL

I file creati sono 3. **CalendarioA.MOD** in cui definiamo il modello vero e proprio con la dichiarazione degli insiemi, delle variabili, dei vincoli e dell'obiettivo. Il file **calendarioA.DAT** contiene invece tutti i valori numerici e i contenuti degli insiemi dichiarati nel file .mod. Questa suddivisione risulta utile per avere maggiore modularità e propensione ad eventuali modifiche ed implementazioni future. Il terzo e ultimo file **calendarioA.RUN** contiene vari comandi fra cui la specifica del risolutore, l'inclusione dei 2 file precedenti e la modalità di visualizzazione a schermo. Viene scritto per comodità e per velocizzare la compilazione dell'applicativo. Le squadre incluse in questa simulazione sono state "solo" 6 (Milan, Inter, Juventus, Napoli, Lazio, Genoa), quindi un dataset ristretto, per comodità e velocità di esecuzione. Sono state incluse 2 squadre aventi lo stesso stadio in comune (Milan, Inter) per un testing più corretto. Viene definito anche un evento "di prova" per occupare uno stadio in una giornata. La prima versione era con 4 squadre ma il risolutore non riusciva a schedulare correttamente le partite secondo i vincoli imposti per un numero così esiguo di partite.

.RUN

Il file inizia con un reset per cancellare eventuali dati rimasti salvati in memoria, dopo di che procede con il caricare in memoria il file .DAT e il file .MOD. Si sceglie il

risolutore **Gurobi** e gli si passa il comando **solve** pe procedere con la risoluzione. Infine, il comando **display** permette la visualizzazione dei risultati. Il comando opzionale **option** permette di impostare diversi metodi di visualizzazione dell'output.

I risultati dell'elaborazione

```

ampl: include calendarioA.run;
Gurobi 5.0.1: optimal solution; objective 120
94 simplex iterations
X [Genoa,*,Marassi,*] (tr)
:   Genoa Inter Juve Lazio Milan Napoli   :=
1   0   0   0   1   0
2   0   1   0   0   0
3   0   0   0   0   0
4   0   0   0   0   0
5   0   0   1   0   0
6   0   0   0   0   0
7   0   0   0   0   0
8   0   0   0   0   1
9   1   0   0   0   0
10  0   0   0   0   0

[Lazio,*,Olimpico,*] (tr)
:   Genoa Inter Juve Milan Napoli   :=
1   1   0   0   0   0
2   0   0   0   0   0
3   0   1   0   0   0
4   0   0   0   1   0
5   0   0   0   0   0
6   0   0   1   0   0
7   0   0   0   0   1
8   0   0   0   0   0
9   0   0   0   0   0
10  1   0   0   0   0

[Milan,*,Giuseppe_Meazza,*] (tr)
:   Genoa Inter Juve Lazio Napoli   :=
1   0   0   0   0   1
2   0   0   0   0   0
3   0   0   0   0   0
4   1   0   0   0   0
5   0   0   0   0   0
6   0   0   0   0   0
7   0   0   0   1   0
8   0   0   1   0   0
9   0   0   0   0   0
10  0   1   0   0   0

[Inter,*,Giuseppe_Meazza,*] (tr)
:   Genoa Inter Juve Lazio Napoli   :=
1   0   0   0   0   0
2   0   1   0   0   0
3   0   0   1   0   0
4   0   0   0   0   1
5   1   0   0   0   0
6   0   0   0   0   0
7   0   0   0   0   0
8   0   0   0   1   0
9   0   0   0   0   0
10  0   0   0   0   0

[Napoli,*,San_Paolo,*] (tr)
:   Genoa Inter Juve Lazio Milan   :=
1   0   0   0   0   0
2   0   0   0   1   0
3   1   0   0   0   0
4   0   0   0   0   0
5   0   0   0   0   0
6   0   1   0   0   0
7   0   0   0   0   0
8   0   0   0   0   0
9   0   0   1   0   0
10  0   0   0   0   1

[Juve,*,Allianz_stadium,*] (tr)
:   Genoa Inter Lazio Milan Napoli   :=
1   0   0   1   0   0
2   0   0   0   0   0
3   0   0   0   0   0
4   0   0   0   0   1
5   0   1   0   0   0
6   0   0   0   0   0
7   1   0   0   0   0
8   0   0   0   1   0
9   0   0   0   0   0
10  0   0   0   0   0
;

```

L'output si presenta in formato tabellare come nell'immagine soprastante. Si è scelto di stampare a video la variabile **X** che è quella che ci fornisce maggiori informazioni. Ad ogni tabella appartiene una squadra con il proprio stadio di

appartenenza. Nella prima colonna sinistra sono rappresentati i numeri delle giornate mentre nella prima riga troviamo tutte le squadre avversarie possibili. La combinazione degli 0 e degli 1 rappresenta l'eventuale presenza (1) o meno (0) dell'incontro in tale giornata.

Conclusioni

Le partite vengono schedate in modo corretto rispettando tutti i vincoli imposti dalle specifiche iniziali. Nei file allegati si è inserito un evento test che occupa nella giornata 3 l'Allianz arena (stadio Juventus) che viene lasciato correttamente libero nella schedulazione finale. Con il risolutore Gurobi ho ottenuto 94 iterazioni con un obiettivo di 120 per arrivare al risultato finale.