

# Metodi e Modelli per l'Ottimizzazione Combinatoria

## Proposta di progetto

Luigi De Giovanni

### 1 Premessa (**REGOLE!!!**)

L'esame di Metodi e Modelli per l'Ottimizzazione Combinatoria consiste di:

- svolgimento di un progetto **individuale** sull'applicazione dei metodi presentati a lezione per la soluzione di un problema di ottimizzazione combinatoria;
- esame orale su tutti gli argomenti del corso.

L'accesso all'esame orale è subordinato alla consegna del progetto e alla sua valutazione positiva da parte del docente. Il voto sarà deciso dal docente in base alla qualità sia del progetto che dell'esame orale in modo paritetico.

Il progetto prevede le seguenti fasi:

1. definizione di un argomento e descrizione del problema a cura dello studente. L'argomento è quello proposto di seguito (a meno di particolari esigenze e proposte che, comunque, vanno prima discusse con il docente). La descrizione deve semplicemente rendere esplicite le richieste del problema in termini di possibili funzioni obiettivo e caratteristiche volute per le soluzioni ammissibili, descritte in linguaggio naturale (1-2 pagine, con eventuali disegni, non si richiede il modello matematico in questa fase);
2. formalizzazione del problema con un modello di programmazione lineare intera mista;
3. implementazione del modello con Cplex e il test su istanze di esempio;
4. implementazione di un metodo di soluzione alternativo, basato su una tecnica di programmazione matematica e/o di una metaeuristica, e test delle tecniche implementate su istanze di prova;
5. la stesura e la consegna di una relazione con la descrizione del lavoro svolto (circa 20 pagine)

A scelta dello studente, il progetto può essere condotto durante lo svolgimento del corso, con delle scadenze fissate per ogni fase. A seconda delle scadenze rispettate e della qualità degli elaborati via via consegnati, saranno dati 5 punti di bonus, da sommare al normale voto dell'esame (progetto + orale). Ovviamente, anche chi non rispetta le scadenze o fa il progetto in altri momenti può prendere trenta e lode, ma è più difficile...

La prima scadenza è fissata per **lunedì 22 ottobre 2012, ore 9:00 A.M.**. Le altre scadenze saranno comunicate durante il corso e pubblicate su questo sito.

**ATTENZIONE!!!** Dopo le scadenze delle fasi 1 e 2, il docente metterà a disposizione degli studenti una descrizione e dei modelli COMUNI a tutti (con qualche eventuale variante individuale), che saranno utilizzati da tutti per le fasi successive, sia chi intende rispettare le scadenze, sia chi non lo farà.

Si propone di seguito una descrizione INFORMALE del problema suggerito per il progetto. Per i dettagli si rimanda a quanto presentato in aula dal Prof. Palazzi e al materiale messo a disposizione su questo sito.

## 2 Progetto proposto: configurazione di reti wireless ad hoc

**ATTENZIONE!** *In questa sezione descriviamo il problema da risolvere. Sulla pagina del corso è anche disponibile del materiale al solo scopo di chiarire il contesto di applicazione. In caso di elementi discordanti, vale quanto detto in questo documento.*

Alcuni dispositivi dotati di sistemi di comunicazione wireless si trovano in una certa area e vogliono stabilire una connessione, finalizzata, per fissare le idee, alla conduzione di un gioco che coinvolga tutti i dispositivi.

Il modello di connessione prevede l'elezione tra i dispositivi di un server con i seguenti compiti:

- raccolta di informazioni sullo stato di tutti i dispositivi;
- elaborazione delle informazioni raccolte;
- distribuzione delle informazioni aggiornate a tutti i dispositivi.

Tutti i dispositivi, compreso il server, agiscono da client: elaborano strategie di gioco e comunicano informazioni di stato al server a intervalli di tempo regolari.

È nota la posizione di ogni dispositivo e, per ciascun dispositivo, la distanza massima entro la quale deve trovarsi un altro dispositivo per poter stabilire una connessione (link).

Per questo motivo, le connessioni da e per il server possono essere multi-hop: in questo caso, i dispositivi intermedi dovranno ricevere e ritrasmettere le informazioni di altri dispositivi inoltrandole verso la destinazione. Si tenga presente che le comunicazioni avvengono punto-punto (non è previsto multi-cast) e che i link hanno una capacità di banda massima, che dipende dalla distanza tra i dispositivi coinvolti.

All'inizio, i dispositivi sono dotati di una carica nota, che viene consumata durante il gioco per i seguenti motivi:

- per i client:
  - consumi della CPU per l'elaborazione delle strategie di gioco;
  - consumi per la trasmissione delle informazioni di stato al server;
  - consumi per la ricezione delle informazioni di stato dal server;
  - eventuali consumi per la ricezione e la ritrasmissione di informazioni di stato provenienti dal server o da altri client.
- per il server:
  - consumi della CPU per l'elaborazione delle strategie di gioco (come client);
  - consumi della CPU per l'elaborazione delle informazioni di stato raccolte;
  - consumi per la ricezione delle informazioni di stato dai client;
  - consumi per la trasmissione delle informazioni di stato aggiornate ai client.

Tutti i consumi sono noti in unità di carica per unità di tempo.

La connessione tra i dispositivi e il gioco si considerano attivi finché tutti i dispositivi sono accesi o, in altri termini, finché uno qualsiasi dei dispositivi esaurisce la batteria; a quel punto il gioco termina, a prescindere da quale dispositivo abbia terminato la sua scorta di energia. Considerando che le diverse comunicazioni hanno durata trascurabile rispetto alla durata della carica di tutti i dispositivi, si vuole determinare la configurazione della rete (scelta del server e routing delle trasmissioni) al fine di massimizzare la durata del gioco.

Per cercare di aumentare la durata del gioco, è possibile considerare un diverso modello di connessione in cui più di un dispositivo, a turno, funge da server. In questo caso, è noto il numero di server e, in ciascun turno, uno solo dei nodi configurati come server è attivo, mentre gli altri sono dormienti. A intervalli regolari, il server attivo diventa dormiente e un server dormiente diventa attivo, secondo una regola round robin. I nodi client, come nel modello precedente, inviano / ricevono le informazioni sullo stato del gioco al / dal solo server attivo. Il server attivo, oltre ai compiti descritti in precedenza, si occupa di comunicare, a intervalli regolari, informazioni di sincronizzazione ai server dormienti (queste informazioni includono, al momento opportuno, anche il comando di cambio turno). Pertanto, oltre ai consumi sopra elencati, si devono considerare:

- consumi del server attivo per la trasmissione delle informazioni di sincronizzazione ai server dormienti;
- consumi dei server dormienti per la ricezione delle informazioni di sincronizzazione;
- consumi dei server dormienti per l'elaborazione delle informazioni di sincronizzazione.

Anche questi consumi sono noti in unità di carica per unità di tempo, e si vuole determinare la configurazione (nodi server in round robin e routing delle comunicazioni) che massimizzi la durata del gioco, sempre considerando che la durata delle diverse comunicazioni è trascurabile.