

Task Assignment distribuito per videosorveglianza - Progetto 6

Cerruti Federico

Fabbro Mirko

Masiero Chiara

23 novembre 2009

1 Formulazione del problema ed obiettivi

Questo lavoro nasce allo scopo di coordinare le attività di una rete di telecamere per la videosorveglianza. L'obiettivo è progettare un algoritmo distribuito efficiente per l'assegnazione di task alle telecamere, che costituiscono un sistema di agenti dotati di risorse limitate, in grado di comunicare almeno parzialmente tra loro. A tal fine, è nostra intenzione analizzare preliminarmente alcune soluzioni al problema del *task assignment* già sviluppate in altri contesti, sia a carattere distribuito, sia centralizzato. Da questo punto di partenza il progetto si svilupperà isolando gli aspetti specifici ed originali che caratterizzano le reti di telecamere e sviluppando, sperabilmente, una o più soluzioni distribuite ad hoc.

Ci si propone di implementare un simulatore per valutare le soluzioni ottenute e confrontarle con il metodo attualmente adottato. Esso consiste in un algoritmo di tipo *brute force*, la cui regola principale è che il task viene assegnato al primo agente che si dichiara libero, mentre le situazioni critiche vengono risolte grazie all'intervento di un *leader*, cioè di un agente privilegiato normalmente mantenuto libero.

Si introduce una prima modellizzazione semplificata della rete di telecamere, utile a cogliere la natura del problema. Esse sono descrivibili come agenti dotati di risorse limitate e si assume, almeno in prima ipotesi, che possano comunicare tra loro (in particolare, il paradigma di comunicazione è del tipo *broadcast pulsato*). Gli agenti sono in grado di svolgere task quali il *tracking*, l'*event detection*, il *bridging* per comunicazioni tra nodi della rete ed il reperimento e stoccaggio di informazioni (ciascun agente può agire da repository per immagini, record sullo stato della rete, etc.).

La richiesta di effettuare il *tracking* giunge tramite la porta seriale, mentre gli input relativi agli altri task e la comunicazione tra gli agenti impegnano la scheda ethernet. La struttura hardware delle telecamere è illustrata in figura 1.

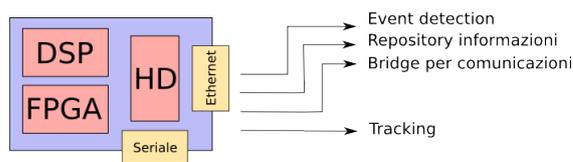


Figura 1: Schema della struttura hardware della telecamera e dei task gestibili

2 Approccio proposto

In assenza di specifiche costrizioni, il problema dell'assegnazione è banale ed è risolvibile affidando ogni nuova istanza ad un agente casuale, scelto tra quelli non occupati. Questo vale, in generale, se ogni task ha la stessa importanza degli altri e se ogni agente ha la stessa attitudine ad eseguire una particolare istanza rispetto agli altri. L'applicazione che si considera, tuttavia, deve occuparsi di richieste profondamente eterogenee tra loro, sia intrinsecamente (i task hanno priorità diverse, richiedono risorse computazionali diverse) sia per ragioni topologiche (il tracking, per esempio, necessita di essere eseguito da un particolare insieme di agenti).

La questione da affrontare, nel modellizzare l'applicazione in forma matematica, è stabilire in maniera univoca una funzione di costo (con struttura il più semplice possibile), che permetta di determinare soluzioni ottime o sub-ottime. Un possibile approccio è la costruzione di una funzione che interpreti, appunto, il costo (o, al contrario, ma in maniera equivalente, l'utilità) associato ad una particolare assegnazione task-agente. Questo indice, che sarà in linea di massima reale e positivo, è interpretabile come metrica nello spazio degli agenti (o dei task). In tal modo si dà luogo ad un'interpretazione geometrica del processo di assegnazione, in cui le camere sono rappresentate da agenti in grado di muoversi nello spazio, verso particolari punti che identificano i task, in maniera istantanea o con traiettorie dinamiche, a

seconda dell'algoritmo considerato.

Algoritmi centralizzati

Si vuole per prima cosa considerare il problema come risolubile in modo centralizzato, quindi con totale disponibilità di informazione ed una singola entità che si fa carico del costo computazionale complessivo. In questa situazione (e sotto ipotesi molto blande) è semplice indicare un criterio di ottimalità per le assegnazioni inteso a minimizzare di una funzione di costo lineare. Tale funzione di costo può essere banalmente identificata come la somma dei pesi c_{ij} dei singoli accoppiamenti task-agente.

$$X = \arg \min_X \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

$$\sum_{i=1}^m x_{ij} \geq 1 \quad \forall j$$

$$x_{ij} = \begin{cases} 1 & \text{se il task } j \text{ è assegnato all'agente } i \\ 0 & \text{altrimenti} \end{cases}$$

Il modello che ne deriva è noto in letteratura come Linear Sum Assignment Problem (LSAP). In [7] sono proposti diversi algoritmi di programmazione matematica in grado di risolvere in tempi polinomiali il LSAP. I risultati ottenuti sfruttando tale approccio combinatorio forniranno, indicativamente, un limite superiore per le prestazioni ottenibili con (ed un termine di confronto per) i successivi algoritmi distribuiti.

Algoritmi distribuiti

In un secondo momento, pur ipotizzando l'informazione una risorsa globale, è necessario considerare logiche distribuite in cui gli agenti decidono per sé stessi a quali task associarsi o, in caso di collisioni, per cui competere. Una famiglia di algoritmi possibile è costituita dagli algoritmi *market-based*, in cui vengono implementati meccanismi ad asta, secondo i quali ogni agente interessato a svolgere un task adotta una particolare politica di rilancio in competizione con gli altri agenti. Ogni task j ha uno specifico valore c_{ij} , dato dalla differenza tra l'utilità a_{ij} per ogni singolo agente ed il suo prezzo p_j .

$$c_{ij} = a_{ij} - p_j$$

Il meccanismo d'asta fa sì che ogni agente insoddisfatto (ovvero cui non è stato assegnato il task con

c_{ij} maggiore) cerchi di scambiare il proprio task con quello di maggior valore, incrementandone il costo globale. L'assegnazione definitiva si determina quando tutti gli agenti sono soddisfatti. Il concetto si può estendere al caso distribuito anche dal punto di vista dell'informazione, dove il prezzo dei task non è una quantità globale.

Un approccio diverso è l'algoritmo di ricerca descritto in [2], per il quale l'assegnazione è assimilabile alla ricerca di target multipli in un ambiente non noto. In questo caso il modello che ne deriva considera l'affioramento dei task in maniera sequenziale e non contemporanea. Tutti gli agenti cercano di associarsi al task attivo e la loro dinamica è descritta dal passo di avvicinamento $\eta(t)$.

$$e_i(t+1) = (1 - \eta(t))e_i(t)$$

$$\eta(t) = \eta_0 \frac{\|e_i(t)\|^{-2}}{\sum_{i=1}^n \|e_i(t)\|^{-2}}$$

dove $e_i(t)$ è il vettore differenza tra il task attivo e l'agente i -esimo. Gli agenti sono in pratica più incentivati ad avvicinarsi tanto più la loro distanza dal task è inferiore rispetto alle distanze degli altri agenti. Le ripetute istanze dei task determinano una dispersione degli agenti che tende, dopo molte ripetizioni, ad una convergenza generale degli stessi sui task. Anche in questo caso risulta cruciale determinare una metrica che sia rappresentativa dell'applicazione reale.

3 Problematiche

L'analisi della natura del problema permette di definire alcune possibili problematiche che potrebbero manifestarsi nelle successive fasi progettuali, anche per effetto della limitazione delle risorse degli agenti sull'assegnazione delle attività. Quando verranno rilasciate le ipotesi semplificative iniziali, dovranno essere affrontate le conseguenze della ridotta disponibilità di informazione accessibile da ogni agente (tipicamente locale) e della non istantaneità della dinamica di rilascio delle risorse (per esempio attraverso la definizione di un *timeout* per misurare l'effetto del tempo di raffreddamento necessario a rendere nuovamente disponibile una risorsa precedentemente impegnata).

Altri aspetti da considerare sono legati alla natura dei task assegnati. Essi potrebbero prevedere l'impegno contemporaneo di più agenti, ad esempio (situazione ben gestibile con tecniche market-based). Inoltre, potrebbe rendersi necessario valutare l'effetto della

disomogeneità dei task, che si manifesta nel caso in cui essi abbiano priorità diverse, come già accennato. Infine, non è da trascurare la valutazione delle conseguenze della rottura di uno o più agenti (che potrebbe essere assimilata ad un impiego ininterrotto di risorse), nonché di eventuali *deadlocks* che potrebbero manifestarsi durante il task assignment.

4 Agenda

Segue una prima pianificazione delle fasi del progetto, soggetta a modifiche in itinere.

- **16/11 - 7/12** Formalizzazione del primo approccio ([2]): modellizzazione e implementazione (metrica, priorità, task multipli)
- **23/11 - 7/12** Ricerca bibliografica e definizione dello stato dell'arte
- **30/11 - 21/12** Implementazione simulatore (in riferimento al primo approccio)
- **30/11 - Gennaio** Analisi di alcuni approcci distribuiti alternativi: market based e derivanti dalla teoria dei giochi
- **Gennaio** Effetti del rilassamento delle ipotesi e analisi della robustezza, con riferimento particolare all'algoritmo presentato in [2].
- **Gennaio-Febbraio** Simulazioni ed ulteriori considerazioni

Riferimenti bibliografici

- [1] Luc Brunet. Consensus-based auctions for decentralized task assignment. Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2008.
- [2] Matthias J. Feiler. On distributed search in an uncertain environment. 2009.
- [3] MyungJoo Ham and Gul Agha. A study of coordinated dynamic market-based task assignment in massively multi-agent systems.
- [4] Nathan Michael, Michael M. Zavlanos, Vijay Kumar, and George J. Pappas. Distributed multi-robot task assignment and formation control.
- [5] Brandon J. Moore and Kevin M. Passino. Distributed task assignment for mobile agents. 2007.
- [6] Diego Pizzocaro. Sensor assignment in a virtual environment using constraint programming. Master's thesis, University of Aberdeen, Department of Computing Science, 2007.
- [7] Burkard R., Dell'Amico M., and Martello S. *Assignment Problems*. SIAM, 2008.