

Lezione 14 — 16 Novembre 2010

Docente: Luca Schenato

Stesori: Guido Albertin, Elena Toffoli, Giancarlo Baldan

14.1 Algoritmi di Consensus

Con il termine consensus si intende la convergenza ad un parametro comune di più agenti tramite scambio di informazioni o interazione locale. Tali algoritmi possono essere applicati efficacemente per risolvere problemi apparentemente molto diversi fra di loro come:

1. rendez-vous (robotica coordinata);
2. stima ai minimi quadrati distribuita;
3. sincronizzazione di orologi;
4. calibrazione di sensori;
5. pattugliamento perimetrale
6. medie generalizzate
7. individuazione di eventi

Alcuni di questi esempi sono illustrati qui di seguito. Ulteriori esempi sono disponibili in F. Garin, L. Schenato. *A survey on distributed estimation and control applications using linear consensus algorithms*. capitolo in *Networked Control Systems*, Springer, 2010 e S. Bolognani, S. Del Favero, L. Schenato, D. Varagnolo. *Consensus-based distributed sensor calibration and least-square parameter identification in WSNs*. *International Journal of Robust and Nonlinear Control*, 2010, disponibili presso la pagina web del corso.

14.1.1 Rendez-vous

Si hanno N veicoli la cui dinamica è data da

$$x_i^+ = x_i + u_i$$

o equivalentemente $x_i^{k+1} = x_i^k + u_i^k$ in cui x_i e u_i sono scalari. Lo scopo è quello di far convergere tutti i veicoli in un unico punto

$$x_i(t) \longrightarrow \bar{x}.$$

Una soluzione banale a questo tipo di problema si ottiene ponendo $u_i = -x_i$ in modo da far convergere tutti gli agenti in un solo passo nell'origine. In termini energetici il punto ideale verso cui convergere è però il centro di massa o comunque un punto interno alla configurazione. In forma vettoriale il problema può essere formalizzato come segue

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^N \quad \mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \in \mathbb{R}^N \quad \mathbf{x}^+ = \mathbf{x} + \mathbf{u}.$$

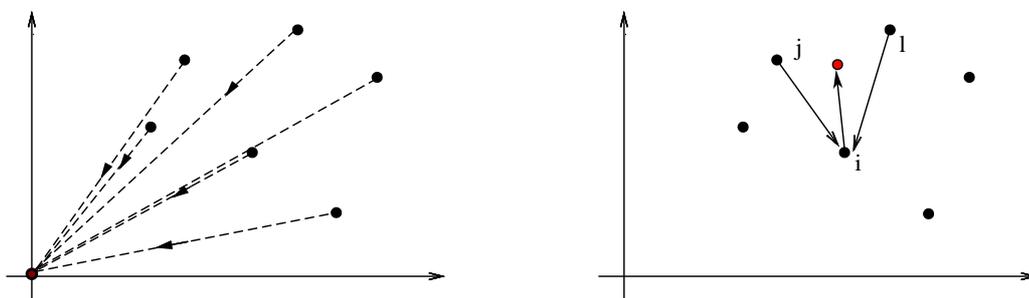


Figura 14.1. A sinistra convergenza in un passo all'origine; a destra l'agente i riceve le posizioni di j e l e si muove di conseguenza.

L'obiettivo è quello di progettare un controllo che porti tutti i veicoli nello stesso punto all'interno della configurazione. L'idea è che ciascun veicolo riceva la posizione di altri agenti presenti nella configurazione e si sposti verso una combinazione convessa delle posizioni stesse

$$\begin{aligned} x_i^+ &= \rho_1 x_i + \rho_2 x_j + \rho_3 x_l = \\ &= (1 - \rho_2 - \rho_3) x_i + \rho_2 x_j + \rho_3 x_l = \\ &= x_i + \underbrace{\rho_2(x_j - x_i) + \rho_3(x_l - x_i)}_{u_i = k_i x} = p_i x \end{aligned}$$

dove i pesi sono $\rho_i \geq 0$ e $\sum \rho_i = 1$. Si osserva che con questo tipo di approccio è necessaria la sola distanza relativa tra gli agenti e non la loro posizione nel piano. Si ottiene così

$$\begin{cases} \mathbf{x}^+ = \mathbf{x} + \mathbf{u} \\ \mathbf{u} = K\mathbf{x} \end{cases}$$

quindi

$$\mathbf{x}^+ = \mathbf{x} + K\mathbf{x} = \underbrace{(I + K)}_P \mathbf{x} \quad P = \begin{bmatrix} p_1 \\ \vdots \\ p_N \end{bmatrix} \in \mathbb{R}^N.$$

La matrice P è una matrice stocastica fatta cioè di elementi non negativi ($p_{ij} \geq 0$) e tale per cui la somma degli elementi di ciascuna riga è unitaria ($\sum_{i=1}^N p_{ij}$). Se l'agente j -esimo

comunica la posizione a quello i -esimo l'elemento p_{ij} sarà positivo mentre in caso contrario esso sarà nullo.

Si può definire a questo punto il vettore

$$\mathbf{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

autovettore destro relativo all'autovalore unitario e tale per cui $P\mathbf{1} = \mathbf{1}$. È proprio grazie a quest'ultima proprietà della matrice P che il punto verso cui i veicoli convergono è un punto di equilibrio (questo non equivale a dire che la matrice P è asintoticamente stabile).

Introduciamo ora

$$S = \left\{ z = \sum_{i=1}^N \alpha_i x_i \mid \alpha_i \geq 0, \sum \alpha_i = 1 \right\}$$

ovvero il poligono convesso (vedi figura 14.2) che include tutti i punti; si può dimostrare che ed ad ogni passo dell'algoritmo di controllo S si riduce o al più resta invariato. Un

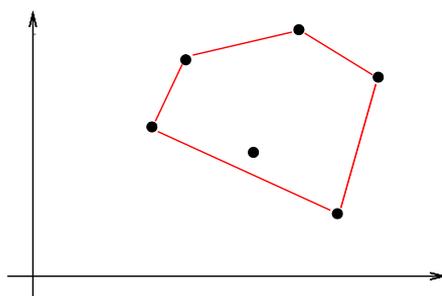


Figura 14.2. Convex hull

altro caso di interesse è quello in cui P è tempo variante; la relazione che lega lo stato della configurazione alle condizioni iniziali ad un generico istante t è esprimibile come

$$x(t) = \left(\prod_{k=1}^t P_k \right) x(0)$$

mentre nel con P costante si aveva $x(t) = P^t x(0)$. Un ulteriore distinzione tra le matrici tempo varianti si può avere in base al metodo con cui vengono scelti i veicoli con i quali ciascun agente comunica ad ogni passo: si distingue tra matrici random (se la scelta avviene in modo casuale) oppure deterministiche.

Bisogna a questo punto dimostrare che S tende a ridursi ad un punto e capire quali siano le proprietà che deve avere la matrice P affinché questo avvenga ovvero sotto quali condizioni

$$P^t \xrightarrow{?} P_\infty \quad \text{oppure} \quad \prod_{k=1}^t P_k \xrightarrow{?} P_\infty.$$

Sarà inoltre importante capire che relazioni intercorrono tra le proprietà della matrice P ed il grafo di comunicazione allo scopo di progettare la matrice di modo che la configurazione converga il più velocemente possibile.

14.1.2 Stima distribuita

Il comando di controllo è in questo caso proporzionale alla stima ottima

$$\mathbf{u} = L\hat{\mathbf{x}} = \begin{bmatrix} l_1 \\ \vdots \\ l_N \end{bmatrix} \hat{\mathbf{x}}.$$

A ciascun agente serve quindi, per comportarsi in modo ottimo, la stima che dipende dalle misure di tutti i sensori essendo $\mathbf{u}_i = l_i \hat{\mathbf{x}}$; tale stima non sarà ovviamente disponibile nel caso in cui il grafo non sia completamente connesso.

Un esempio può essere quello in cui ciascuno degli N nodi misura la temperatura all'interno di una stanza

$$y_i = T + \nu_i \quad \text{dove} \quad \nu_i \sim N(0, 1).$$

La stima ottima di massima verosimiglianza date tutte le misure è la media

$$\hat{T}_{glob} = \frac{1}{N} \sum_{i=1}^N y_i$$

ed il nostro obiettivo è quello di trovare un'algoritmo di controllo tale per cui

$$\hat{T}_i(t) \longrightarrow \hat{T}_{glob}.$$

L'idea è di utilizzare come stima iniziale la misura di ciascun sensore $\hat{T}_i(0) = y_i$ e di iterare l'algoritmo

$$\hat{T}^+ = P\hat{T}.$$

Nel caso in cui $P^t \rightarrow P_\infty$ e imponendo che P sia anche stocastica per colonna ovvero che $\mathbf{1}^T P = \mathbf{1}^T$ varrà anche $\mathbf{1}^T P^t = (\mathbf{1}^T P)P^{t-1} = \mathbf{1}^T$. Supponendo inoltre che P_∞ sia tale per cui

$$\mathbf{x}(t) = P^t \mathbf{x}(0) \quad \longrightarrow \quad P_\infty \mathbf{x}(0) = \bar{x} \mathbf{1}$$

dalle seguenti relazioni

$$\begin{aligned} \mathbf{1}^T \mathbf{x}(t) &= \mathbf{1}^T P^t \mathbf{x}(0) & \longrightarrow & \quad \mathbf{1}^T P_\infty \mathbf{x}(0) = \mathbf{1}^T \mathbf{x}(0) \\ \mathbf{1}^T \bar{x} \mathbf{1} &= \mathbf{1}^T \mathbf{x}(0) \end{aligned}$$

segue

$$\bar{x} = \frac{1}{N} \sum_i x_i(0)$$

ovvero se si ha convergenza il valore verso cui tutti i sensori tendono è la media stessa, ma solo nel caso in cui P sia doppiamente stocastica. Si avrà in oltre

$$\hat{T}(t) \longrightarrow \frac{1}{N} \sum_i y_i \mathbf{1} = \hat{T}_{glob} \mathbf{1}.$$

14.1.3 Sincronizzazione

Si hanno in questo caso a disposizione N orologi ciascuno dei quali fornisce una misura locale del tempo

$$\tau_i(t) = \alpha t + o_i.$$

Si muovono tutti alla stessa velocità α ma partendo da istanti diversi o_i . Lo scopo è quello di realizzare uno stimatore funzione del tempo locale ed in grado di compensare l'offset iniziale

$$\hat{\tau}_i(t) = f(\tau_i) + \hat{o}_i(t) = t + o_i + \hat{o}_i(t)$$

di modo che dopo un certo numero di iterazioni gli orologi restituiscano la stessa stima

$$\hat{\tau}_i(t) \longrightarrow \tau_{glob} \quad \forall i$$

con $\tau_{glob}(t) = t + o_{glob}$. Ponendo $x_i(t) = o_i + \hat{o}_i(t)$ ci si può ricondurre ad un problema di consenso infatti

$$t + o_i + \hat{o}_i(t) \longrightarrow t + o_{glob}$$

diventa ora

$$x_i(t) \longrightarrow o_{glob}.$$

A questo punto si tratterà di scegliere una matrice stocastica P tale per cui tutte le $x_i(t)$ convergano ad un unico valore (non necessariamente la misura esatta del tempo). In analogia a quanto visto in precedenza si avrà quindi

$$\mathbf{x}^+ = P\mathbf{x}$$

In forma vettoriale

$$\hat{\mathbf{o}} = \begin{bmatrix} \hat{o}_1 \\ \vdots \\ \hat{o}_N \end{bmatrix} \in \mathbb{R}^N \quad \mathbf{o} = \begin{bmatrix} o_1 \\ \vdots \\ o_N \end{bmatrix} \in \mathbb{R}^N \quad \mathbf{x}^+ = \mathbf{o} + \hat{\mathbf{o}}$$

$$(\mathbf{o} + \hat{\mathbf{o}})^+ = P(\mathbf{o} + \hat{\mathbf{o}}) = (I + K)(\mathbf{o} + \hat{\mathbf{o}}).$$

Quindi

$$\begin{aligned} \hat{\mathbf{o}}(t+1) &= \hat{\mathbf{o}}(t) + \mathbf{o} - \mathbf{o} + K(\mathbf{o} + \hat{\mathbf{o}}) = \\ &= \hat{\mathbf{o}}(t) + K(\mathbf{o} + \hat{\mathbf{o}}) \end{aligned}$$

come in precedente la matrice K sarà facilmente ricavabile dalla matrice P (e viceversa) sulla base della relazione $P = I + K$; si osserva in oltre che essa, anche in questo caso, introduce fattori moltiplicativi le differenze tra gli offset iniziali ed il termine atto a compensarli. In altre parole per ciascuna componente si avrà $[K\mathbf{x}]_i = \sum_j \rho_i(x_j - x_i)$ dove $x_j - x_i$ può essere espresso

$$\underbrace{t + o_i + \hat{o}_i(t)}_{\hat{\tau}_i(t)} - \underbrace{(t + o_j + \hat{o}_j(t))}_{\hat{\tau}_j(t)}.$$

Si può osservare come in questo caso le stime $\hat{o}_i(t)$ tendano a valori diversi essendo

$$\hat{\mathbf{o}}(t+1) = \hat{\mathbf{o}}(t) + K\hat{\boldsymbol{\tau}}(t)$$

mentre a convergere sono $o_j + \hat{o}_j(t)$. Il problema può essere esteso anche al caso in cui i sensori si muovano a velocità diverse.

14.1.4 Calibrazione

Si hanno ora N sensori che raccolgono misure affette da offset

$$y_i = \Theta + o_i$$

Per calibrazione si intende il problema di stimare ed eliminare tale offset in modo da fornire una misura accurata. Cioè di vorrebbe calcolare una stima \hat{o}_i tale che

$$\hat{y}_i = y_i - \hat{o}_i \approx \Theta$$

In mancanza di un nodo calibrato non è possibile pensare di poter compensare esattamente tutti i sensori, poichè al massimo è possibile calcolare la differenza tra due offset calcolando le differenze tra misure di sensori diversi, cioè $y_i - y_j = o_i - o_j$. È possibile però fare in modo che tutti i nodi abbiano lo stesso offset, cioè progettare un algoritmo tale che $o_i - \hat{o}_i = \alpha$ per tutti i nodi. Questo in fatti si traduce in un problema di consensus sulla variabile $x_i(t) = o_i - \hat{o}_i(t)$. In particolare se scegliamo di implementare un algoritmo di consensus che garantisce la convergenza alla media delle condizioni iniziali $x_i(0)$ ed inizializziamo le stime iniziali a zero $\hat{o}_i(0) = 0$ si ottiene che

$$o_i(t) \rightarrow o_i - \frac{1}{N} \sum_{i=1}^N o_i$$

Se gli offset sono distribuiti in modo tale da avere media nulla, per esempio come nel caso di $o_i \sim \mathcal{N}(0, \sigma^2)$, allora $\frac{1}{N} \sum_{i=1}^N o_i \rightarrow 0$ per $N \rightarrow +\infty$ e quindi $\hat{o}_i(t) \rightarrow o_i$. Quindi sebbene non sia possibile cancellare completamente l'offset, tale errore desce a zero se il numero di nodi è molto grande.