11ème Atelier en Evaluation de Performances
Toulouse, du 15 au 17 mars 2016

Compter avec les Protocoles de Population

Yves Mocquard

Université de Rennes 1
Irisa
yves.mocquard@irisa.fr

1. Introduction

Le modèle des protocoles de population, introduit par Angluin et al. [1], fournit les bases théoriques pour analyser les propriétés émergeant d'un système constitué d'agents anonymes interagissant deux à deux. Nous définissons dans [2], un protocole de population permettant de compter exactement le nombre d'agents d'un type particulier. Nous montrons que ce protocole converge de manière logarithmique en temps distribué.

Beaucoup de travaux existent sur le problème de la majorité, Angluin et al. [4] ont mis au point un protocole rapide et simple, mais qui n'est exact que quand la majorité est large, ça reste un bon outil pour trouver un consensus. Alistarh et al. [4] ont récemment mis au point un protocole de majorité exact en en temps logarithmique mais avec une constante très grande.

Nous allons décrire un protocole de population qui résout le problème du comptage, lequel généralise celui de la majorité en l'affinant. En interrogeant n'importe quel agent, on peut connaître le nombre d'agent d'un type particulier en $O(\log n)$ interactions, avec une probabilité aussi grande que l'on souhaite.

2. Protocoles de Population

La définition qui suit est tirée de Angluin et al. [5].

Un protocole de population est caractérisé par un 6-uplet $(Q, \Sigma, Y, \iota, \omega, f)$, Σ est l'ensemble fini des symboles d'entrée, Y est l'ensemble fini des symboles de sortie, $\iota : \Sigma \to Q$ est la fonction d'entrée qui détermine l'état initial d'un agent, $\omega : Q \to Y$ est la fonction de sortie qui détermine le symbole de sortie d'un agent, et $f : Q \times Q \to Q \times Q$ est la fonction de transition qui décrit comment deux agents interagissent et mettent à jour leur état.

Au début, tous les agents démarrent avec un symbole initial venant de Σ. La fonction ι initialise l'état de chaque agent à partir de son symbole, puis au gré des interactions, les agents mettent à jour leur état utilisant la fonction de transition f.

Soit $C = \{C_t, \ t \geq 0\}$ un processus stochastique à temps discret avec comme ensemble d'états Q^n. Nous l'appellerons configuration un état de ce processus stochastique pour ne pas le confondre avec l'état des agents du protocole de population. Pour tout $t \geq 0$, la configuration à l'instant t de ce processus stochastique est notée par $C_t = (C_t^{(1)}, \dots, C_t^{(n)})$, $C_t^{(i)}$ représente l'état de l'agent i à l'instant t.

A chaque instant t, deux indices distincts i et j sont successivement choisis parmi $1, \dots, n$ avec la probabilité $p_{i,j}(t)$. Nous nommons X_t la variable aléatoire représentant ce choix, c'est-à-dire

$$\mathbb{P}\{X_t = (i, j)\} = p_{i,j}(t).$$

Nous considérons que les variables aléatoires X_t et C_t sont indépendantes.

3. Compter avec les Protocoles de Population

3.1. Introduction

Chaque agent possédant un symbole d'entrée issu de l'ensemble $\Sigma = \{A, B\}$, soit N_A et N_B le nombre d'agents ayant pour symbole respectivement A ou B, le but est de connaître la différence invariante $\kappa = N_A - N_B$. Et c'est ce que doit rendre la fonction de sortie ω.

Au début la fonction ι attribut à chaque agent l'état m ou $-m$ selon que son symbole soit respectivement, A ou B.

La fonction de transition consiste à attribuer à chaque agent la moyenne des deux valeurs.

Cet algorithme garde constant la somme de tous les états de chaque agent, somme qui est égale à $m(N_A - N_B)$.

La fonction de transition égalise progressivement l'état de tous les agents qui finissent par approcher de la même valeur $\frac{m(N_A - N_B)}{n}$. A partir de laquelle la fonction de sortie peut calculer $\kappa = N_A - N_B$.

3.2. Compter avec un ensemble d'états fini

Nous travaillons avec un ensemble fini d'états Q qui est un ensemble d'entier. Les paramètres Q, Σ, Y, ι et ω dépendent de l'application et seront définis à la fin de cette section pour le calcul de la différence invariante κ. La fonction de transition f est définie par

$$f(a, b) = \begin{cases} \left(\frac{a+b}{2}, \frac{a+b}{2}\right) & \text{si } a + b \text{ est pair} \\ \left(\frac{a+b-1}{2}, \frac{a+b+1}{2}\right) & \text{si } a + b \text{ est impair} \end{cases}$$

Une fois que le couple (i, j) est choisi à l'instant t, C_{t+1} est défini par

$$\left(C_{t+1}^{(i)}, C_{t+1}^{(j)}\right) = \begin{cases} \left(\frac{C_t^{(i)}+C_t^{(j)}}{2}, \frac{C_t^{(i)}+C_t^{(j)}}{2}\right) \text{ si } C_t^{(i)} + C_t^{(j)} \text{ pair} \\ \left(\frac{C_t^{(i)}+C_t^{(j)}-1}{2}, \frac{C_t^{(i)}+C_t^{(j)}+1}{2}\right) \text{ si } C_t^{(i)} + C_t^{(j)} \text{ impair} \end{cases}$$

$$\text{et } C_{t+1}^{(m)} = C_t^{(m)} \text{ pour } m \neq i, j.$$

Le lemme suivant est fondamental, il établit que la somme des états des agents reste invariante.

Lemme 1 *Pour tout* $t \geq 0$*, nous avons*

$$\sum_{i=1}^{n} C_t^{(i)} = \sum_{i=1}^{n} C_0^{(i)}.$$

Nous notons ℓ la moyenne des coordonnées de C_t et L le vecteur de $\mathbb{R}^n$ dont toutes les coordonnées sont égales à ℓ, c'est-à-dire

$$\ell = \frac{1}{n} \sum_{i=1}^{n} C_t^{(i)} \text{ et } L = (\ell, \ldots, \ell).$$

Théorème 2 *En supposant un choix uniforme du couple* (i, j)*, c'est-à-dire si, pour* $i \neq j$*,*

$$p_{i,j}(t) = \frac{1}{n(n-1)},$$

alors nous avons

$$\mathbb{E}\left(\|C_t - L\|^2\right) \leq \left(1 - \frac{1}{n-1}\right)^t \mathbb{E}\left(\|C_0 - L\|^2\right) + \frac{n}{4}.$$

En utilisant l'inégalité de Markov on obtient le corollaire suivant qui donne une δ-approximation de l'écart maximum entre les coordonnées de C_t et ℓ.

Corollaire 3 *Pour tout* $\delta \in]0, 1[$*, s'il existe une constante* K *telle que* $\mathbb{E}(\|C_0 - L\|_\infty) \leq K$ *alors, pour tout* $t \geq (n-1)\ln(4K^2)$ *nous avons*

$$\mathbb{P}\{\|C_t - L\|_\infty \geq \sqrt{\frac{n}{2\delta}}\} \leq \delta.$$

Nous pouvons maintenant appliquer ces résultats au calcul de la différence invariante κ. L'ensemble des entrées est $\Sigma = \{A, B\}$, et la fonction d'entrée ι est définie par $\iota(A) = m$ et $\iota(B) = -m$, m étant un entier strictement positif. Cela signifie que, pour tout $i = 1, \ldots, n$, $C_0^{(i)} \in \{-m, m\}$. Nous avons

$$\ell = \frac{1}{n} \sum_{i=1}^{n} C_0^{(i)} = \frac{\kappa m}{n},$$

ce qui montre à partir du Lemme 1 que κ est invariant. L'ensemble des états Q est maintenant l'ensemble $\{-m, -m+1, \ldots, m-1, m\}$. La fonction de sortie est, pour tout $x \in Q$,

$$\omega(x) = \lfloor nx/m + 1/2 \rfloor.$$

L'ensemble de sortie Y est l'ensemble des valeurs possibles de κ, i.e. $Y = \{-n, -n+1, \ldots, n-1, n\}$.

Théorème 4 *Pour tout* $\delta \in]0, 1[$*, en prenant* $m = \left\lceil \sqrt{2} n^{3/2} / \sqrt{\delta} \right\rceil$ *et pour tout* $t \geq (n-1)\left(5\ln 2 + 3\ln n - \ln \delta + \frac{2}{m-1}\right)$*, nous avons*

$$\mathbb{P}\{\omega(C_t^{(i)}) = \kappa, \text{ pour tout } i = 1, \ldots, n\} \geq 1 - \delta.$$

Donc le temps de convergence pour obtenir κ avec une probabilité aussi grande que l'on veut est $O(n \log n)$ et ainsi le temps de convergence parallèle pour obtenir κ avec une probabilité aussi grande que l'on veut est $O(\log n)$.
Ce protocole nécessite la connaissance préalable du nombre d'agents n.
Dans le futur nous comptons proposer un protocole qui n'a plus cette nécessité. Deux voies sont possibles, la première consiste à calculer la différence en terme de pourcentage, la deuxième consiste, en utilisant une élection de leader, à calculer, au préalable, la taille du système.

Bibliographie

1. Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fisher, and René Peralta. Conputation in netWorks of passively mobile finite-state sensors. *Distributed Computing*, 18(4):235-253,2006.
2. Yves Mocquard, Emmanuelle Anceaume, James Aspnes, Yann Busnel, and Bruno Sericola. Counting with Population Protocols. In *Proceedings of the 14th IEEE International Symposium on Network Computing and Applications (IEEE NCA15)*, September 2015.
3. Dan Alistarh, Rati Gelashvili, and Milan Vojvonic. Fast and exact majority in population protocols. Technical report, Microsoft Research, 2015.
4. Dana Angluin, James Aspnes, and David Eisenstat. A simple population protocols for fast rubust approximate majority. 20(4) :279-304, 2008.
5. Dana Angluin, James Aspnes, David Eisenstat, and Eric Ruppert. The computationnal power of population protocols. *Distributed Computing*, 20(4) :278-304,2007.