Allocation de ressources par une méthode hybride machine learning - optimisation dans un contexte de conteneurs

Etienne Leclercq[1,2], Jonathan Rivalan[1], Frédéric Roupin[2], Céline Rouveirol[2]

[1] Alter way, Saint-Cloud, France
{etienne.leclercq,jonathan.rivalan}@alterway.fr
[2] LIPN, UMR 7030, Université Paris 13, Sorbonne Paris Cité
{leclercq,roupin,rouveirol}@lipn.univ-paris13.fr

Mots-clés : *optimisation, clustering, cloud computing, allocation de ressources.*

1 Introduction

Alter way, une entreprise spécialisée dans les plateformes web, migre actuellement depuis ses solutions d'hypervision d'un système en VMs vers un système en conteneurs, en amenant de nouvelles constraintes pour le problème d'allocation de ressources, comme une consommation de ressources ératique dûe à l'élasticité des conteneurs. Bien que l'optimisation des ressources en contexte de conteneurs est aujourd'hui très étudiée (voir [1]), comme ce fut le cas pour les machines virtuelles (VMs) (voir [2]), la technologie des conteneurs reste relativement nouvelle dans les systèmes en production et l'on manque encore d'intelligence dans les techniques d'allocation associées.

L'objectif de ce travail est d'obtenir une plateforme optimisée pour la gestion des conteneurs, en anticipant au mieux les incidents de production et en économisant des ressources. Là où l'apprentissage par renforcement a été utilisé pour améliorer la qualité de la solution [3], l'originalité ici est d'utiliser l'apprentissage pour identifier des profils de consommation en vue de guider le processus d'optimisation.

2 Approche considérée

Notre méthodologie se divise en plusieurs étapes, avec tout d'abord une phase d'apprentissage non supervisé dans laquelle nous cherchons à identifier des profils de consommation au niveau des conteneurs, grâce aux traces de consommation passées. Ceci est réalisé par un clustering de séries temporelles (une série par historique de consommation par conteneur), dont la première implémentation utilise un algorithme de k-means.

L'étape suivante est d'utiliser ces profils pour guider la solution de notre problème d'allocation de ressources. Afin d'anticiper au mieux d'éventuels incidents de production, nous préférons avoir des noeuds avec une consommation globale la plus stable possible. Pour cela nous avons implémenté une heuristique qui prend les deux clusters les plus éloignés par leur centre, et assigne les conteneurs par paire - un de chaque cluster - sur un noeud, jusqu'à ce qu'on atteigne la capacité du noeud pour remplir de la même manière le noeud suivant. Ceci est répété jusqu'à épuisement des clusters, en reprenant les deux clusters suivants les plus distants. En faisant cela, nous visons à regrouper les conteneurs avec des profils *opposés*, en vue d'obtenir une consommation totale *régulière* sur les noeuds.

La dernière étape, non explicitée ici, consiste à évaluer l'allocation proposée, afin d'adapter le clustering si besoin et donc d'améliorer la solution finale.

Le mécanisme global est illustré en Fig. 1.

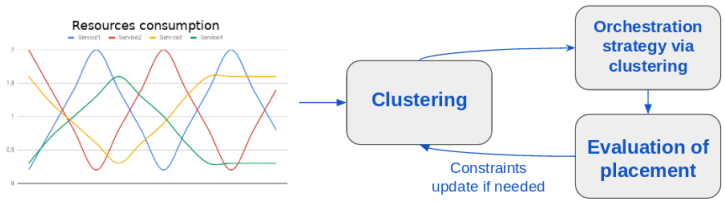
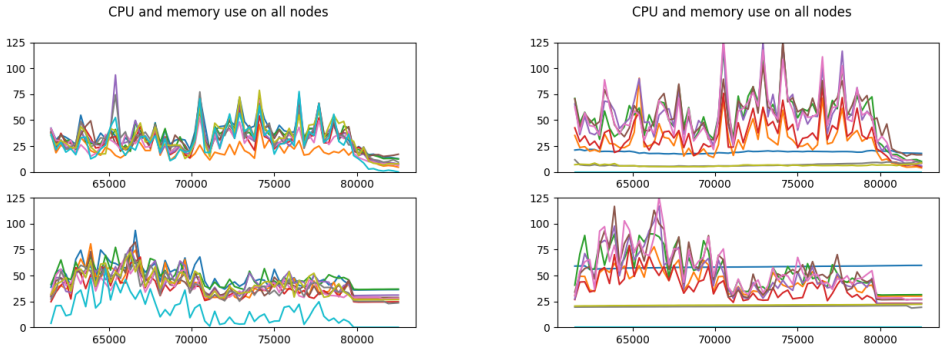


FIG. 1 – Schéma de la méthodologie



(a) Placement initial

(b) Placement via notre méthode

FIG. 2 – Consommation totale de chaque noeud sans notre méthode (a) et avec notre méthode (b) en fonction du temps (timestamp en secondes) sur la deuxième période, après clustering ($k = 3$) sur la première période.

3 Tests expérimentaux

Une première version simplifiée a été testée sur traces historisées venant du cloud d'Alibaba [4], sur lesquelles nous avons d'abord divisé en deux la période de temps disponible dans les traces : la première pour identifier des profils de consommation (via clustering), la deuxième pour effectuer notre heuristique de placement et comparer avec le placement initial. La comparaison d'évolution de consommation sur les noeuds (voir Fig. 2) montre d'une part l'économie d'un noeud (sur 10 initialement), et d'autre part une hétérogénéité quant à la régularité de la consommation globale de chaque noeud.

Nos premiers résultats sur données réelles montrent des résultats encourageants sur certains aspects, tout en soulignant que la méthodologie - en particulier les différentes briques de celle-ci - est perfectible. Il s'agit en effet d'un travail préliminaire et nous projetons d'explorer plusieurs méthodes de clustering, plusieurs heuristiques d'allocation pour améliorer les économies, ou encore envisager la ré-allocation (en cas d'application de politiques d'allocation plus risquées par exemple).

Références

[1] Andrea Tosatto, Pietro Ruiu, and Antonio Attanasio. Container-based orchestration in cloud : State of the art and challenges. *2015 Ninth International Conference on Complex, Intelligent, and Software Intensive Systems*, pages 70–75, 2015.

[2] Zoltán Ádám Mann. Allocation of virtual machines in cloud data centers - a survey of problem models and optimization algorithms. *ACM Comput. Surv.*, 48 :11 :1–11 :34, 2015.

[3] A. Galstyan, K. Czajkowski, and K. Lerman. Resource allocation in the grid using reinforcement learning. In *AAMAS '04 Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, volume 3, pages 1314–1315, 2004.

[4] Alibaba. Alibaba cluster trace program, 2017. Available at https ://github.com/alibaba/clusterdata.