

Émulation du scheduling de Linux dans Batsim

Contexte

Batsim est un simulateur d'infrastructures informatiques de calcul distribué spécialisé sur l'étude de politiques de gestion de ressources. Une partie importante des politiques des gestion de ressources est l'algorithme de *scheduling* qu'elles utilisent. Batsim est un programme en C++ construit au-dessus de la bibliothèque de simulation SimGrid, dont Batsim utilise les modèles de calcul et réseau. Batsim est généralement utilisé par son interface réseau (une socket ZeroMQ), ce qui permet d'y connecter des codes de prise de décision et de contrôle de la simulation, quel que soit le langage utilisé pour développer ces codes. Batsim a été initialement conçu afin d'étudier des politiques de gestion de centres de calcul à haute performance. Batsim fournit différents modèles afin qu'une utilisatrice puisse définir le comportement de ses applications.

Le découplage entre simulateur et code de prise de décision de Batsim a permis d'étudier des implémentations réelles de gestionnaires de ressources comme OAR, Slurm, Flux ou encore Kubernetes. Tous ces gestionnaires de ressources prennent des décisions au niveau de toute une plateforme distribuée.

Objectifs du projet

Nous voulons dans ce projet étendre les cas d'utilisation de Batsim afin qu'il permette d'étudier des politiques de décision à grain très fin. Linux est une famille de systèmes d'exploitation libre dont le noyau manipule en permanence des processus. Les algorithmes de scheduling de Linux doivent décider très souvent et en un temps extrêmement court quels processus placer sur les cœurs de la machine.

L'objectif principal de ce projet est de permettre la connexion des **vrais** codes des algorithmes de scheduling de Linux à Batsim, afin que ces algorithmes puissent être étudiés en simulation dans Batsim. Plus précisément, voici une liste indicative des tâches de ce projet :

- Prendre en main Batsim et le scheduling dans Linux.
- Modéliser une instance d'exécution simple de processus Linux en tant que *workload* Batsim.
- Étudier en détail le code de scheduling de Linux puis connecter ce code à Batsim.
- Réaliser une comparaison de différents algorithmes de Linux par simulation Batsim.