Communications réseau pour apprentissage automatique dans un environnement distribué, hétérogène et volatile

Sujet de stage I3 / T3 / Master 2

Contexte

Hive est une entreprise qui propose aux particuliers comme aux entreprises de mettre à disposition leurs ressources informatiques inutilisées. Hive propose ainsi une offre de stockage de données, HiveDisk, qui utilise les espaces de stockage accordés par les contributeurs à HiveDisk. Cela permet aux utilisateurs de HiveDisk de profiter d'un stockage géo-distribué et répliqué. De la même façon, Hive souhaite pouvoir partager avec HiveCompute les ressources de calculs (surtout des GPUs) inutilisées pour réaliser principalement des tâches d'entraînement et d'inférence d'applications d'intelligence artificielle. L'utilisateur demande sur une interface web d'obtenir un certain nombre de GPUs, situés sur différentes machines, et peut ensuite y accéder pour exécuter ses calculs. Dans un premier temps, les GPUs alloués seront sur des machines appartenant à un même réseau local (par exemple le réseau d'un site d'entreprise), mais l'objectif à terme est de pouvoir utiliser des GPUs situés sur différents réseaux d'entreprises à différents endroits (par exemple tous les sites d'une entreprise à l'échelle d'un pays).

Ce projet fait face à de nombreux défis, surtout venant du fait que l'environnement ciblé est différent des environnements traditionnellement utilisé pour des calculs d'entraînement ou d'inférence, tels que le HPC ou le cloud. D'un point de vue matériel, les machines sont moins puissantes, hétérogènes et sont interconnectées par un réseau classique, moins performant qu'un réseau HPC. Il faut également prendre en compte que les ressources de calculs ne sont pas disponibles en permanence (par exemple, les machines sont moins disponibles en journée car les employés les utilisent) et qu'elles sont plus susceptibles de disparaître à tout moment. De plus, utiliser des machines appartenant à différents sites géographiques crée un réseau aux performances hétérogènes : la latence pour communiquer entre deux sites est bien plus élevée qu'au sein d'un même site.

Objectifs du stage

Ce stage se concentre sur les problématiques liées aux communications réseau dans un tel contexte. Les objectifs de ce stage sont donc les suivants :

- réaliser un état de l'art des solutions existantes pour pouvoir exécuter des applications d'apprentissage dans un tel contexte (géo-)distribué, hétérogène et volatile ;
- passer à la pratique : choisir une application de machine learning et essayer de l'exécuter dans un contexte matériel et logiciel ressemblant à celui de HiveCompute ;
- expérimenter les différentes bibliothèques de communications possibles (MPI, des versions de MPI plus adaptées à ce genre d'environnement, `libp2p`, *etc*) et l'interfaçage possible avec les bibliothèques d'apprentissage tels que PyTorch.

Profil recherché

Compte tenu des objectifs du stage, il est préférable que le/la stagiaire soit à l'aise dans un environnement Linux, avec les langages C et Python et en programmation réseau et système. Des compétences en HPC et apprentissage automatique seraient un plus.

Modalités du stage

Encadrement

- Philippe Swartvagher – `philippe.swartvagher@inria.fr`
- Thomas Hérault – `herault.thomas@gmail.com`

Lieu du stage

Centre Inria de l'Université de Bordeaux
Équipe-projet Topal
200, avenue de la Vieille Tour
33405 Talence

Gratification

4,35 €/h selon la réglementation sur la gratification minimale, *cf* le site du ministère pour plus de renseignements. Cela représente environ 600 €/mois.

Poursuite en thèse

Ce stage a pour vocation de déboucher sur une thèse, dont le financement est en cours d'acquisition.

Une fois ce modèle de communication réseau le plus adapté défini, cette thèse s'intéressera, étant donné un ensemble de machines et leur topologie, aux adaptations nécessaires aux schémas de communications des applications d'apprentissage pour minimiser le coût des communications : par exemple en utilisant des algorithmes de routage et une répartition des calculs et des données plus adaptés au réseau connectant les machines. Il faudra

également être en mesure de détecter la disparition et l'ajout possible de machines et s'adapter en conséquence, par exemple en ignorant les contributions des machines perdues dans le cas d'un parallélisme de données, ou bien en redistribuant les données et les calculs. On pourra également considérer la gestion de l'occupation du réseau dans le cas où HiveDisk et HiveCompute sont présents simultanément sur les mêmes réseaux et les mêmes machines, afin de conserver des performances satisfaisantes pour les deux services et adapter dynamiquement les paramètres de qualité de service en fonction des conditions du réseau et des exigences des utilisateurs.