contribution des réacteurs, mais aussi celle des mouvements d'air turbulents autour de l'appareil. Or pour une simulation adaptée, il faudrait compter environ une semaine de calcul sur une machine de quelque 500 000 processeurs dédiés. Pure théorie, car le plus gros calculateur installé en France, la machine Pangea chez Total Exploration Production, n'a que 18 400 processeurs de ce type. Sur cette machine, la simulation d'un avion requerrait des mois voire des années de calcul. Impensable !

La future voiture autonome illustre quant à elle les besoins en traitement de gros volumes de données (HPDA). Toutes les 90 minutes, un tel véhicule générera environ 4 To de données par jour, venues des radars, sonars, GPS, lidars, caméras embarquées... En déplacement dans un environnement complexe, la voiture devra traiter les données en temps réel, et prendre des décisions immédiates afin d'éviter des accidents. Il lui faudra donc aussi une capacité de calcul importante à bord.

VERS LA CONVERGENCE

Ces exemples montrent que les attentes en capacités de calcul sont importantes, surtout dans une société où le *big data* sera un carburant central (*voir l'entretien avec D. Cohen, page 8*). Les efforts de recherche pour les améliorer sont à la hauteur des attentes, et l'on pourrait même assister à une convergence entre HPC et HPDA. Pour quelles raisons ?

Les gros centres de calcul, autrefois dédiés au HPC et équipés pour effectuer des opérations très nombreuses sur des données souvent peu nombreuses, ont cependant des capacités de stockage très importantes. Or le HPDA est justement le traitement de gros volumes de données. De plus, outre l'infrastructure de stockage des données, les centres de calcul ont également celle nécessaire à leur traitement, même si parfois les capacités requises pour les gros volumes de données du HPDA sont légèrement différentes des capacités de traitement pour le HPC.

Plus précisément, un centre de calcul HPC dispose d'un espace de stockage rapide des données, de nombreux nœuds de calcul, d'un système d'interconnexion rapide entre ces nœuds ainsi qu'avec le système de stockage. Les nœuds sont par ailleurs équipés pour faire du calcul décimal très efficacement (pour la résolution d'équations mathématiques bien souvent).

Puisque les infrastructures pour le HPC sont en place, il est tentant de leur faire exécuter des programmes de traitement de type HPDA, puisque tous les ingrédients existent et permettent d'effectuer les traitements HPDA. Il suffirait de les adapter en augmentant l'espace de stockage des données et en améliorant l'efficacité des interconnexions entre les nœuds de calcul et avec le stockage. En fin de compte, la convergence HPC-HPDA permettrait de mieux utiliser les ressources matérielles existantes.

On observe également un rapprochement des communautés HPC et intelligence artificielle, notée IA, car elles exécutent les mêmes types d'applications intensives en données ou en calcul que le HPDA, que ce soit sur de très gros supercalculateurs, sur des petits groupes institutionnels de machines (des *clusters*), ou dans le nuage (on parle alors de *cloud computing*).

L'ENVOL DE L'IA

La révolution de l'IA, par laquelle les ordinateurs arrivent à créer des modèles prédictifs à partir de données (*voir La révolution de l'apprentissage profond, par Y. Bengio, page 42*), conduit à l'adoption rapide de cette technologie HPC qui a également rendu possibles de nombreux développements scientifiques, algorithmiques, logiciels et matériels.

Le fait que la communauté IA utilise l'infrastructure HPC est un élément important dans la convergence HPC-HPDA, car le matériel, très cher, est déjà en place et suffisant pour du traitement HPDA. Les *data scientists* n'ont donc pas à investir dans du matériel, tandis que les utilisateurs du HPC peuvent maximiser l'utilisation de leurs ressources en les mettant à la disposition d'autres applications.

On peut en conséquence s'attendre à un essor de l'IA : les machines nous aideront de plus en plus à prendre des décisions complexes. Pour ce faire, des jeux de données de plus grande taille et plus complexes doivent être utilisés lors de la phase d'apprentissage de l'IA. L'IA sera présente dans les voitures autonomes, >

À quoi ressemble l'atmosphère des exoplanètes ? RobERt vous répond. Il a appris à les sonder grâce au *deep learning*.

mais aussi en science par exemple pour identifier des cellules cancéreuses sur des images biomédicales, trouver des événements clés en physique des hautes énergies, repérer des événements biomédicaux rares…

Afin que les scientifiques puissent extraire de la richesse des volumes énormes de données mesurées, modélisées ou simulées, l'évolutivité est la clé. Cette évolutivité consiste à répartir les calculs sur un grand nombre de processeurs (ou nœuds de calcul), et donc d'effectuer les calculs plus rapidement ou bien de traiter des volumes de données plus importants. Les programmes doivent alors être capables de tirer parti d'autant de processeurs que possible.

MONTE-CARLO, PATRIE DU HASARD

Par exemple, dans la communauté HPC, les scientifiques ont tendance à se concentrer sur la modélisation et la simulation avec des techniques telles que Monte-Carlo afin de produire des représentations précises de ce qui se passe dans la nature. Rappelons que les méthodes Monte-Carlo sont une famille de procédés algorithmiques visant à calculer une valeur numérique approchée en utilisant des techniques probabilistes. L'idée essentielle est d'utiliser le hasard pour résoudre des problèmes qui seraient déterministes en principe. Ils sont souvent utilisés dans des problèmes physiques ou mathématiques et sont plus utiles lorsqu'il est difficile voire impossible d'utiliser d'autres approches.

Les méthodes de Monte-Carlo sont intéressantes, car elles peuvent être parallélisées, et sont donc théoriquement rapides (chaque élément fourni par le processus probabiliste étant indépendant, les calculs peuvent se faire sur autant de processeurs que disponibles). Cependant, elles ne convergent que si la taille de l'échantillonnage est suffisamment grande. Chaque processeur doit donc traiter séquentiellement plusieurs éléments, augmentant d'autant le temps de calcul total.

En pratique, les physiciens des hautes énergies et les astrophysiciens examinent des images à identifier grâce à des techniques d'apprentissage profond (le *deep learning*), c'est-à-dire en les comparant à celles d'une base de données importante. Pour ce faire, les images de référence ont été étiquetées avec des informations les décrivant, puis livrées à l'ordinateur pour qu'il apprenne à « voir » ce qu'elles représentent. Ainsi éduquée, la machine peut reconnaître ce que montrent des images inconnues (on parle d'inférence).

Toujours en astrophysique, le programme RobERt (Robotic Exoplanet Recognition) traque la composition de l'atmosphère des exoplanètes. Mis au point à l'University College de Londres, il fonctionne grâce à une phase d'apprentissage à partir de larges bases de données non structurées recueillies par les instruments de mesure (*voir la figure page précédente*).

Dans d'autres domaines, l'apprentissage consiste à examiner des séries, faire du traitement du signal, des analyses harmoniques, de la modélisation et de la prédiction avec des systèmes non linéaires…

De même, l'IA tente de générer des représentations précises de ce qui se passe dans la nature à partir de données non structurées recueillies par des capteurs, des caméras, des microscopes électroniques, des séquenceurs… La nature inconnue de ces données non structurées conduit les *data scientists* à passer une grande partie de leur temps à extraire et nettoyer les informations utiles pour entraîner le système IA.

Les programmes écrits en langage productif, tel Julia, parviennent rarement à tirer parti du supercalculateur

La convergence HPC-IA concerne aussi bien les algorithmes qui doivent s'adapter à la taille des données et à celle de l'ordinateur cible, que la façon dont les calculs sont répartis sur plusieurs processeurs, le matériel, le logiciel, le prétraitement des données. Les langages de programmation à haute productivité sont également un moteur de rapprochement. De tels langages, comme Python et Julia, permettent de développer des programmes rapidement, car ils sont plus intuitifs dans leur écriture. L'objectif est que les utilisateurs puissent implémenter leurs modèles de calcul, indépendamment du volume de données disponible et de la plateforme de calcul cible.

Soulignons que les seules mesures de performance qui comptent pour la phase d'apprentissage du *deep learning* sont le temps nécessaire pour obtenir le modèle, et la précision de ce dernier. On sait depuis des décennies que cette étape d'apprentissage se comporte de façon quasi linéaire en fonction du nombre de nœuds de calcul, c'est-à-dire que le temps de