LA RECHERCHE

M 1108 - 204 - 32 F

mensuel n° 204 novembre 1988 - 32 fra

SPECIAL

RCCHBV 11 (204) 1285-1428 (1988) ISSN 0029-5671

LES NOUVEAUX ORDINATEURS

BELGIQUE : 234 FB CANADA : 5,50 $ ESPAGNE : 650 PTAS SUISSE : 9,50 FS MAROC : 23 DH TUNISIE : 1.800 MIL SÉNÉGAL : 1.500 CFA USA : NYC 2.95 $ other : 3

Les supercalculateurs à transputers

par Traian Muntean

« Il est européen et révolutionnaire » *affirmait, en 1983, la firme britannique Inmos lors de la première présentation de son transputer, un composant d'un genre nouveau tout spécialement conçu pour s'intégrer dans des machines massivement parallèles, autrement dit des calculateurs abritant une myriade de processeurs coopérant à l'exécution de la même tâche. Réunissant, sur une même puce, tous les ingrédients nécessaires au calcul et au stockage des données mais aussi à la communication avec ses homologues, le transputer a également le mérite de pouvoir être produit en grande série, pour un prix fort raisonnable. Du coup, il est aujourd'hui utilisé dans le cadre de nombreux projets devant mener au développement de calculateurs puissants mais relativement peu coûteux. Mieux : certains de ces projets ont d'ores et déjà abouti en donnant naissance à des produits bel et bien commercialisés.*

Pour simuler le comportement d'un avion en vol ou celui d'un véhicule dans une collision, pour interpréter des images complexes, concevoir des circuits intégrés sophistiqués, étudier le comportement d'une molécule ou d'un réseau de neurones, les industries aéronautiques, spatiales et automobiles, électroniques et nucléaires, les centres de recherche en télécommunications ou en biotechnologies — pour ne citer qu'eux — sont de plus en plus gourmands en puissance de calcul, avides d'ordinateurs toujours plus performants. Mais l'architecture des supercalculateurs d'aujourd'hui (dérivant du principe du « séquentiel », autrement dit le traitement d'une seule opération à la fois) est difficilement extensible. La puissance de ces machines, d'ailleurs fort coûteuses, augmentera encore — de nombreux modèles sont d'ores et déjà capables d'effectuer quelques opérations simultanément et leurs unités de calcul, autrement dit leurs processeurs, seront encore améliorées —, mais pas indéfiniment. On a donc songé à construire des supercalculateurs capables d'effectuer un grand nombre d'opérations à la fois, en parallèle, sur des informations (des données) multiples. Cette nouvelle classe de machines, les calculateurs « parallèles » voire « massivement parallèles », (le nombre de processeurs étant facilement extensible), réunissant une myriade d'unités de calcul élémentaires homogènes ou non, coopérant à l'exécution d'une même tâche, doivent permettre d'obtenir des performances équivalentes voire, à terme, supérieures à celles des supercalculateurs « classiques ». Leur coût, en revanche, devrait être moindre, grâce à l'utilisation de processeurs plus lents certes, mais aussi moins sophistiqués, pouvant donc être produits en grande série. Notons, en outre, que les architectures de ces machines parallèles présentent l'avantage d'être conçues pour répondre au mieux à une large gamme de besoins, puisque le nombre d'unités de calcul qu'elles contiennent peut être adapté à chaque cas. Cette approche devrait permettre de donner naissance à des machines modulaires, allant du simple poste de travail individuel puissant (encore appelé station de travail) au véritable supercalculateur.

Un effort considérable de recherche et de développement est encore nécessaire pour acquérir une parfaite maîtrise de telles machines et accroître la connaissance pratique nécessaire à leur utilisation. Car leur mise au point, comme leur mise en œuvre, est complexe. Le tout n'est pas de connecter de multiples processeurs entre eux — tâche en elle-même éminemment délicate, mais surtout coûteuse, si l'on songe que l'interconnexion totale de mille processeurs nécessiterait un million de fils, ce qui n'est pas envisageable... —, encore faut-il les synchroniser et leur permettre d'échanger efficacement entre eux des informations. De nombreux laboratoires, de par le monde, cherchent à trouver le compromis idéal, autrement dit la forme géométrique d'interconnexion, qui minimiserait le nombre de liaisons entre processeurs sans trop accroître la distance qui les sépare (autrement dit le nombre de connexions qui les séparent) tout en permettant à l'utilisateur de « croire » que chaque processeur est directement connecté à chaque autre. Mais le problème n'est pas seulement architectural, il est aussi logiciel — les langages informatiques couramment utilisés aujourd'hui ont été conçus pour des machines séquentielles, pas pour des calcutateurs parallèles — et matériel : les microprocesseurs, ces unités de calcul intégrées sur une seule puce, présentent l'avantage d'être fabriqués en grande quantité et donc à faible prix, mais n'ont pas été pensés pour fonctionner en parallèle.

On rêve depuis longtemps d'un composant qui serait aux machines massivement parallèles ce que le transistor est aux composants électroniques : une brique de base, peu coûteuse, produite en grandes séries mais parfaitement adaptée, réunissant, en l'occurrence, les ingrédients essentiels pour s'intégrer dans une architecture parallèle. La question, qui ne date pas d'hier, a su être concrétisée outre-Manche. En 1979, une petite équipe de chercheurs embauchés par la toute nouvelle firme britannique Inmos (citons de mémoire, après Iann Barron, co-fondateur d'Inmos, D. May, I. Pearson, R.Taylor, C. Whitby-Strevens), forte des avancées scientifiques en matière de programmation dite parallèle (suite, notamment, aux travaux de l'équipe du professeur C.A.R. Hoare à l'université d'Oxford), entame l'étude d'un composant d'un nouveau genre, destiné à travailler en groupe, autrement dit en réseau. Ambitieux, voire révolutionnaire il y a dix ans, dans cette Europe placée sous la tutelle technologique américaine, ce programme donne naissance en 1985 au premier Transputer (la première version, une puce de 45 mm^2, fut présentée pour la première fois à la fin de l'année 1983 au Salon international des composants de Paris). Depuis, le transputer — dont nous supprimons volontairement la majuscule — est devenu un nom générique recouvrant une famille entière

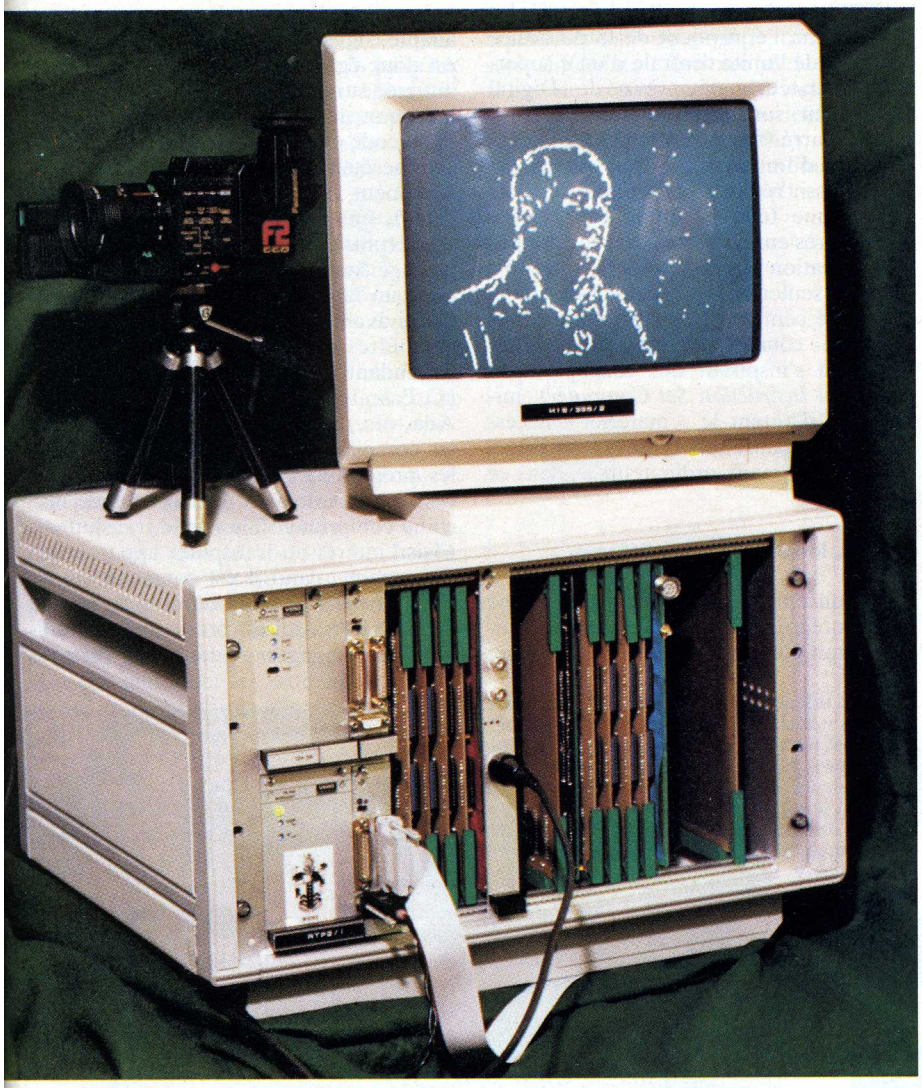


Figure 1. Projet franco-britannique mené sous la bannière du programme Esprit, SuperNode a d'ores et déjà donné naissance à plusieurs prototypes, dont nous voyons ici l'un des premiers exemplaires. Ces machines dites parallèles et reconfigurables doivent permettre de fournir une puissance équivalente à celle des supercalculateurs classiques pour un coût moindre. Elles font appel aux transputers d'Inmos, des composants spécialement développés pour s'intégrer dans ce type d'architectures. Celle du SuperNode est particulièrement originale (voir la figure 6) : reconfigurable, elle permet à son utilisateur de disposer d'une machine dont l'organisation interne s'adapte au mieux au problème à traiter, quel que soit son type. (Cliché Projet Esprit)

de composants programmables particulièrement intégrés, qui peuvent être utilisés comme processeurs indépendants mais qui, surtout, peuvent être interconnectés en nombre important, autorisant ainsi la réalisation efficace (et à faible coût) de systèmes parallèles. Ils possèdent bien les éléments essentiels pour être la brique dont on fait les machines parallèles, l'équivalent du transistor pour les composants électroniques.

Le « transistor » des machines parallèles.

Chaque transputer peut réunir, sur la même puce, une ou plusieurs unités de calcul, de la mémoire et des liens d'interconnexion rapides lui permettant d'échanger des données avec d'autres transputers, voire d'autres composants environnants. Avant d'entrer plus avant dans les détails, précisons la signification de quelques termes nécessaires à la compréhension de la suite. Rappelons, notamment, que toute information, c'est-à-dire toute donnée traitée par un ordinateur, est représentée de façon binaire, l'unité élémentaire de ce code étant le « bit » (contraction de l'expression anglo-saxonne *binary digit*) dont la valeur ne peut être que 0 ou 1 — rappelons également qu'un « octet » est un groupe de huit bits. Le cœur de toute machine de traitement de l'information est constitué de son (ou ses) unité(s) arithmétique et logique (que plus généralement et par extension nous appellerons indifféremment ici un processeur central). Le rôle de cette unité est d'exécuter, comme son nom l'indique, des opérations arithmétiques et logiques qui lui sont dictées par des « instructions ». Les valeurs sur lesquelles ces opérations sont effectuées, et leurs résultats immédiats, sont stockés dans la mémoire centrale de la machine. On mesure donc les performances du processeur central en Mips, c'est-à-dire en millions d'instructions propres exécutées par seconde. Ces instructions, nous l'avons vu, peuvent correspondre à des opérations diverses. Les plus complexes (et les plus longues à exécuter) sont les opérations en virgule flottante, qui permettent d'effectuer des calculs sur de très grands ou très petits nombres avec la même précision — un avantage cher aux scientifiques —, les performances des machines se mesurant alors en Mflops, millions d'opérations en virgule flottante par seconde *(floating point operations per second* en anglais).

Le premier membre de la famille des transputers, le T414, abrite un processeur central capable de traiter des données sur 32 bits, 2 kilo-octets de mémoire interne (permettant de stocker, par exemple, environ 2 000 caractères), une interface rapide lui permettant éventuellement d'accéder à une plus large mémoire externe et quatre liens de communication indépendants pouvant échanger des données à la vitesse maximale de vingt millions de bits par seconde (20 Mbits/s). La simplicité étant le maître mot de la conception, la vitesse de communication entre les transputers est optimisée par l'utilisation de fils unidirectionnels « série » (les bits sont transmis un à un pour simplifier la connectique), chacun reliant uniquement deux transputers. Le T414 permet ainsi d'atteindre une puissance de traitement de l'ordre de 10 millions d'instructions par seconde (10 Mips) pour un programme séquentiel (nous verrons plus loin que l'une des originalités du transputer est d'avoir été conçu pour pouvoir exécuter des programmes parallèles sur un seul processeur).

Le dernier-né de la série, le T800, a été conçu par Inmos dans le cadre d'une collaboration franco-britannique menant au développement d'un supercalculateur européen (fig. 1): le projet SuperNode du programme Esprit — *European Strategic Programme for Research and Development in Information Technology* — de la CEE. Le T800 tient un peu plus de place sur le silicium (20 % seulement), mais il intègre en supplément une unité de calcul en virgule flottante, contrairement aux microprocesseurs classiques à qui il est nécessaire d'adjoindre un autre circuit — voire plusieurs — pour réaliser avec la même précision des opérations arithmétiques sur de très grands ou de très petits nombres. Le T800, à lui seul, présente des performances supérieures à la plupart des microprocesseurs largement commercialisés, mais ce gain est évidemment plus apprécié lorsque l'on connecte dans un même système un grand nombre de transputers.

Le transputer est original parce qu'il comprend — nous l'avons vu — des uni-

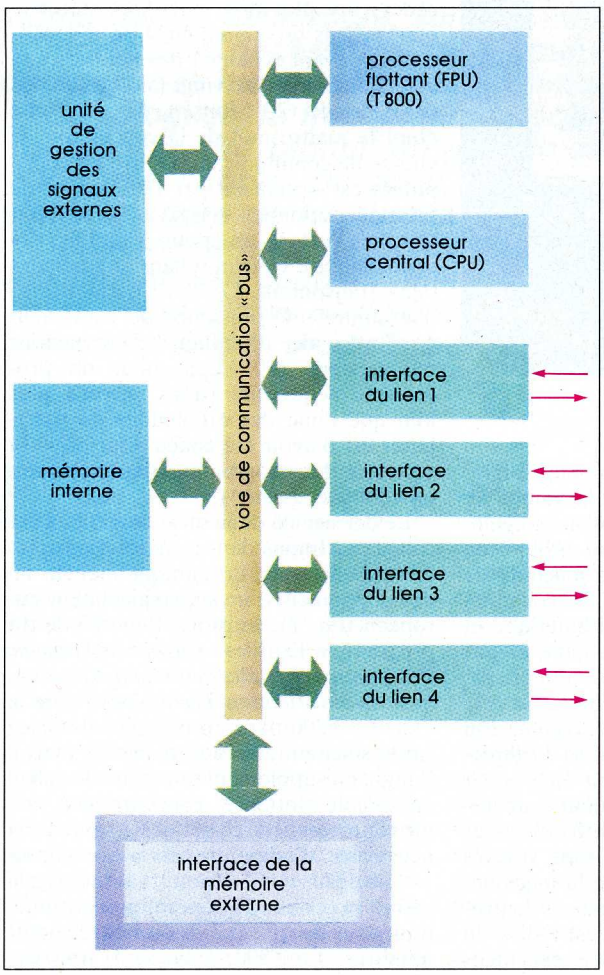
Traian Muntean est directeur de recherches au laboratoire de génie informatique de l'université de Grenoble (IMAG). Il est un des responsable du projet SuperNode du programme Esprit, dont le but est de développer un supercalculateur européen hautement parallèle et reconfigurable à base de transputers.

Le transputer : une « puce » spécialement conçue pour travailler en groupe.

tés de calcul et une mémoire interne (les données sont plus rapidement accessibles lors des opérations élémentaires que si elles étaient stockées dans une mémoire externe) mais aussi quatre liens lui permettant de dialoguer en direct avec autant de ses homologues (fig. 2 et 3). Il est encore bien plus original, parce qu'il autorise ses six « moteurs » (six dans un T800) à tourner en même temps : le processeur central, l'unité virgule flottante, les quatre voies de communication peuvent fonctionner simultanément. Ainsi le T800-30 (le plus puissant des T800) permet à la fois d'effectuer plus de deux millions d'opérations arithmétiques en virgule flottante par seconde (2 Mflops), dix millions d'instructions machine par seconde et de transmettre vingt millions de bits par seconde sur chacun des quatre liens (le T414 atteint seulement cent mille opérations en virgule flottante par seconde, car ces opérations, exécutées dans un T800 sur l'unité de calcul supplémentaire, sont simulées par programme sur l'unique unité de calcul du T414). Bref, un T800 représente, toutes choses égales par ailleurs, l'équivalent de la puissance de calcul de l'unité centrale d'un « superminicalculateur » VAX 8600 de Digital Equipment, sur une seule puce d'un centimètre carré (fig. 2). Comment les développeurs d'Inmos ont-ils pu, sur un espace aussi réduit, bâtir une machine offrant une telle performance ? Grâce aux progrès enregistrés par les techniques de fabrication des composants, bien sûr, mais pas seulement. L'architecture même de l'unité centrale du transputer est très simple — comme ses concepteurs l'ont voulu en s'inspirant des concepts RISC *(Reduced Instruction Set Computer)*, minimisant d'autant le « matériel » nécessaire à sa réalisation (voir « L'architecture des nouveaux ordinateurs », dans ce numéro).

Occam, langage du parallélisme.

La conception d'un composant destiné à être intégré dans une architecture parallèle aurait été incomplète sans la mise au point d'un langage de programmation adapté. Très logiquement, le transputer est donc également la première machine intégrée sur une puce à avoir été spécialement conçue pour exécuter avec l'efficacité du code machine (que l'on peut définir comme étant les ordres élémentaires directement compréhensibles par la machine), un langage parallèle de haut niveau (plus synthétique, plus proche du langage humain), nommé Occam. C'est donc un langage qui a été formellement défini avant le transputer. Le transputer peut être utilisé comme processeur indépendant en utilisant d'autres langages (C, Pascal, Fortran, Lisp, Prolog, bientôt Ada, etc.) pour lesquels on a développé les compilateurs nécessaires (c'est-à-dire les programmes permettant de traduire ces langages de haut niveau en code machine compréhensible par le transputer). Mais l'intérêt du transputer apparaissant clairement quand il est connecté en réseau, la mise au point d'outils destinés à faciliter le travail du programmeur utilisant ces langages — dans des versions



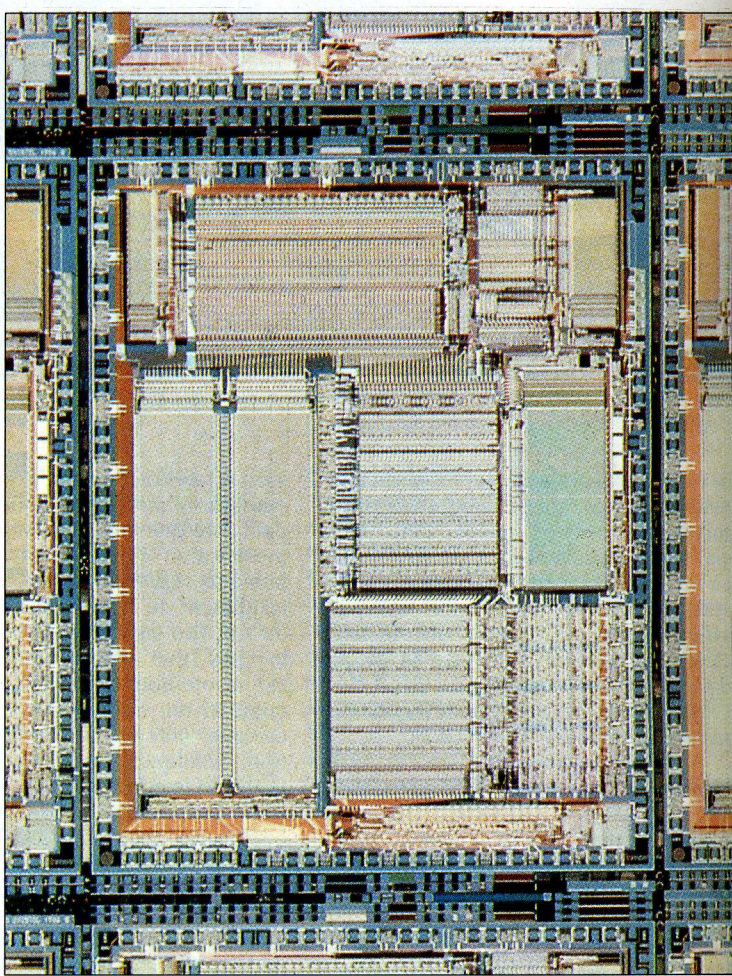


Figure 2. La série des transputers développés par la firme britannique Inmos est bâtie sur une même architecture de base, schématisée ici. Elle est caractérisée par la présence d'une ou deux (deux dans le cas du T800 que l'on voit sur la photographie) unités de calcul (le CPU et le FPU en haut à droite sur le shéma), d'une mémoire interne, d'une unité de gestion des signaux externes (horloge, analyse, erreur etc.) (en haut à gauche) et de quatre liens (fléches rouges) permettant à ce composant original de dialoguer simultanément avec autant de ses homologues : le transputer est un véritable ordinateur intégré sur une seule puce, tout spécialement conçu, de surcroît, pour travailler en groupe. Il a été étudié pour exécuter sans peine des programmes spécifiquement écrits pour des architectures parallèles (réunissant un grand nombre de processeurs coopérant à la même tâche). La deuxième unité de calcul — le FPU — du T800 (correspondant, sur la photo, au bloc supérieur), l'autorise à effectuer, avec la même précision, des opérations sur de très petits ou de très grands nombres, et le rend donc particulièrement bien adapté aux calculs scientifiques. (Cliché Inmos)

spécialement étudiées pour fonctionner sur des architectures à haut degré de parallélisme — est actuellement la priorité de nombreux projets faisant appel, à l'échelle industrielle, aux transputers (non seulement en Europe mais aussi, plus récemment, aux Etats-Unis et au Japon).

Occam est un langage de programmation parallèle de haut niveau issu du modèle de programmation parallèle CSP *(Communicating Sequential Processes)* développé initialement à l'université d'Oxford par l'équipe de C.A.R. Hoare. Le nom du langage n'a pas été choisi au hasard : il reflète la volonté de procurer simplicité et efficacité dans toutes les opérations occupant le plus fréquemment un microprocesseur au cours de l'exécution des programmes. Il s'inspire de la doctrine d'un philosophe franciscain, Guillaume d'Ockham (1270-1349 ?), théoricien du « nominalisme » (une doctrine philosophique plus connue par son Principe du Rasoir : *« Entia non sunt multiplicanda praeter necessitatem »* — il ne faut pas multiplier les entités au-delà de ce qui est nécessaire). Notons que, bien qu'il ait été initialement développé comme langage de programmation de la famille des transputers d'Inmos, Occam peut être également utilisé pour d'autres machines. Il présente, en outre, une remarquable simplicité syntaxique. Occam n'est pas seulement un langage de programmation, mais un véritable formalisme pour la construction correcte des systèmes parallèles, selon des règles précises. En utilisant les outils appropriés, on peut déterminer si un programme Occam est « vrai » ou « faux » (comme tout énoncé logique), autrement dit s'il est exécutable correctement ou non. Ce souci de développement formel a été le fil conducteur dans toute la conception des transputers et d'Occam.

Occam permet d'exprimer un programme comme une collection de processus concurrents — pouvant s'exécuter sur un seul ou sur plusieurs transputers en parallèle. Le processus est un concept fondamental hérité de CSP de Hoare. Dix ans après les premiers pas vers le modèle de programmation parallèle CSP, Hoare propose dans son ouvrage *Communicating Sequential Processes* d'oublier un instant les ordinateurs et leur programmation, et de penser plutôt aux objets qui nous entourent, qui agissent et interagissent avec nous et entre eux, en accord avec un modèle caractéristique de comportement. Pour décrire ce modèle de comportement, il faut tout d'abord décider quels types d'événements ou d'actions seront significatifs et choisir un nom pour chaque type. Prenons le cas d'une simple machine à boissons, pour laquelle il existe deux types d'événements : « pièce », correspondant à l'insertion d'une pièce dans l'orifice de la machine, et « choc », représentant la distribution de chocolat par la machine. Une mémoire d'ordinateur peut aussi être représentée suivant un type de comportement mais les types d'événements qui la caractérisent sont différents : écriture et lecture. Le choix de ces événements implique en général une simplification délibérée, la décision d'ignorer certains aspects, présentant un intérêt moindre, du comportement d'un objet (la taille du gobelet par exemple). C'est ce modèle d'abstraction que nous retrouverons dans le langage Occam où un processus est perçu, dans certains cas, comme une boîte noire cachant les interactions internes des processus composants. Car le processus, entité modulaire de base du langage, peut être formé d'une instruction unique, d'un groupe d'instructions ou d'un groupe d'autres processus communicants ou concurrents.

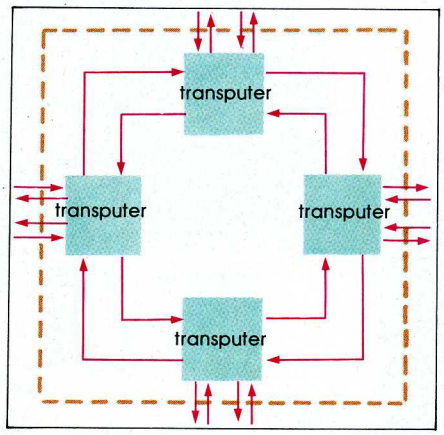


Figure 3. Chaque transputer étant doté de quatre liens (devant lui permettre de dialoguer aisément avec quatre de ses homologues), il est fort simple de réaliser un réseau, comme ici, comprenant quatre transputers partiellement ou totalement interconnectés : ce réseau peut, à son tour, être fonctionnellement considéré comme un « super-transputer » (à huit liens sur la figure), plus puissant, que l'on pourra de la même manière interconnecter avec d'autres. Et ainsi de suite... Les liens des transputers sont constitués de deux fils unidirectionnels rapides (un pour chaque sens) : l'information y transite à la cadence maximale de 800 kilo-octets (sur le modèle T414) ou de 1,7 méga-octet par seconde sur le dernier-né de la famille, le T800.

Objets, messages et primitives.

Un processus Occam est une entité (un objet) qui échange des informations avec son environnement, formé d'autres entités du même type, par l'intermédiaire de « messages » envoyés à travers des « canaux » constituant de véritables contacts unidirectionnels reliant les processus deux à deux. Ce protocole de base de communication entre processus Occam est reproduit, au niveau matériel, par les liens des transputers. Des schémas plus complexes de communication et de synchronisation (des processus plus complexes) peuvent être obtenus par des « constructions » — nous expliquerons plus loin ce terme.

Chaque processus Occam a une existence dans le temps : il commence à un instant précis, exécute un certain nombre d'opérations à une vitesse propre, puis il se termine (correctement ou non). Chaque opération, nous l'avons vu, peut consister en une instruction élémentaire, en plusieurs instructions, ou en une construction d'un nouveau processus à partir d'un ensemble de processus à exécuter séquentiellement, en parallèle ou en « exclusion mutuelle » (un processus, parmi plusieurs). Le canal est le seul élément de communication entre deux processus s'attendant mutuellement. L'un joue le rôle d'« émetteur », l'autre celui de « récepteur » et c'est au cours d'un rendez-vous qu'ils se synchronisent et échangent un message.

Conformément au postulat médiéval précité, les programmes Occam sont construits à partir d'un nombre réduit d'instructions élémentaires qui constituent des processus fondamentaux. Ces briques de base à partir desquelles sont élaborés des processus plus complexes sont au nombre de trois et correspondent au minimum d'opérations nécessaires au fonctionnement de la machine : calcul très élémentaire et échange de messages. L'instruction « affectation » — en simplifiant outrageusement — permet d'assigner une certaine valeur à une place donnée dans la mémoire centrale, l'instruction « entrée » exprime l'attente d'un message sur un canal, et l'instruction « sortie » (le processus complémentaire d'« entrée ») permet l'envoi d'un message sur un canal.

Les processus élémentaires cités ci-dessus peuvent être combinés pour former des constructions séquentielles (grâce à un opérateur appelé SEQ), parallèles (opérateur PAR) ou « alternatives » (opérateur ALT) — l'alternative permet, par exemple, à un processus d'attendre des messages sur différents canaux ou jusqu'à un certain temps propre. Toute construction est elle-même un processus qui peut être un composant d'une autre construction. Occam comprend en outre deux instructions que l'on retrouve dans tout langage de programmation séquentiel : IF (structure conditionnelle) et WHILE (structure itérative).

La répartition des processus concurrents s'exécutant en parallèle sur un réseau de transputers peut prendre plusieurs formes. Un système de processus concurrents peut, en effet, être physiquement réalisé par un réseau de transputers dans lequel chacun d'entre eux exécute un processus, les communications entre les processus étant supportées par les liens physiques des transputers. Mais cette solution (un processus par transputer) reste économiquement coûteuse. Pour la plupart des applications, on préfère donc utiliser une machine comprenant des transputers interconnectés selon une forme géométrique (une « topologie ») donnée, sur laquelle on répartit les processus concurrents de façon optimale. Dans ce cas, chaque transputer est amené

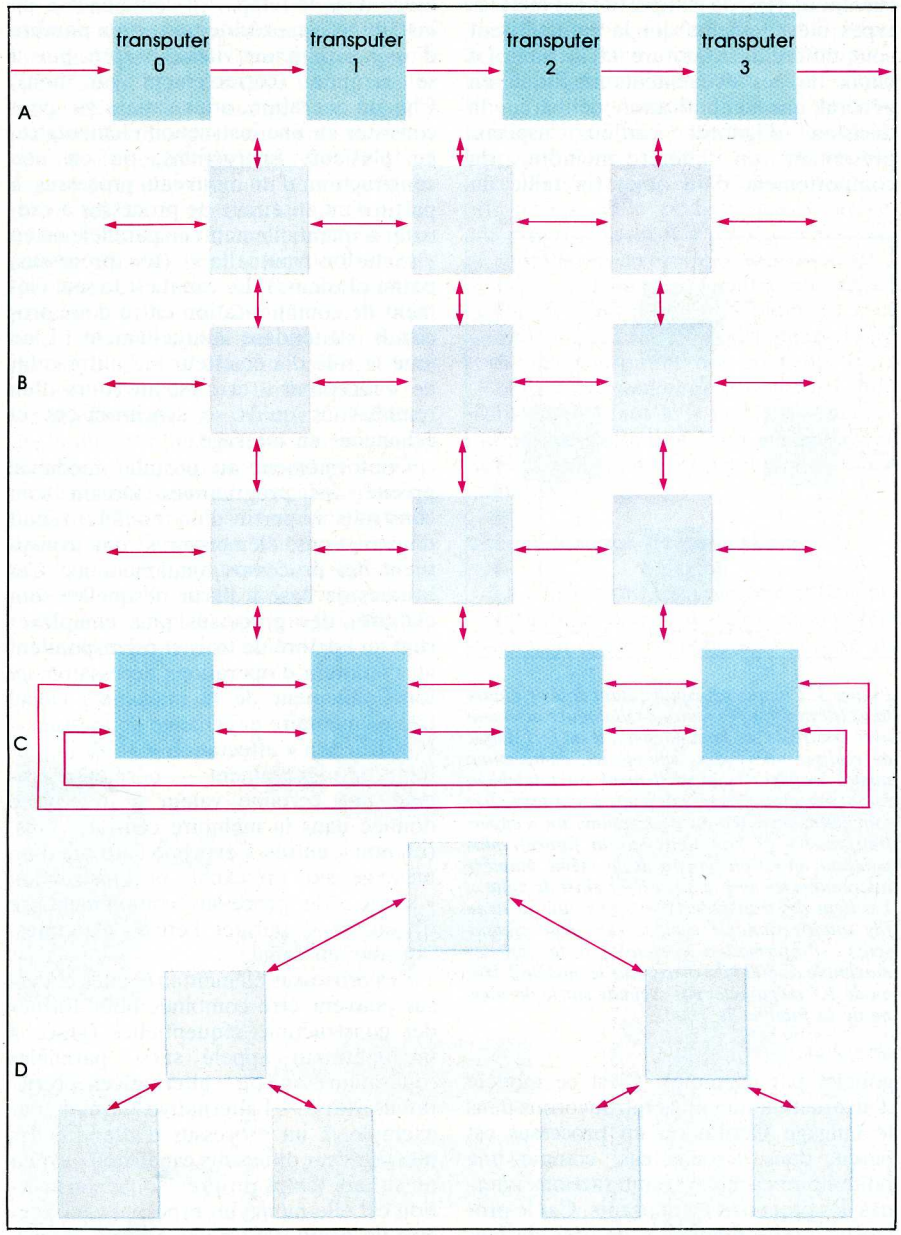


Figure 4. Pour relier entre elles plusieurs unités de calcul (ou processeurs) coopérant à la même tâche, afin de construire des machines dites parallèles, les architectes de l'informatique ont imaginé des réseaux d'interconnexion de formes géométriques très variées. Le parallélisme le plus courant (celui que l'on rencontre dans tous les processeurs des supercalculateurs classiques actuellement commercialisés) est connu sous le nom de « pipeline » (A) : une tâche est décomposée en plusieurs étapes que l'on appellera des sous-tâches, exécutées en cascade par autant d'unités de calcul spécialisées — ici, des transputers. Les transputers sont tout spécialement conçus, au contraire de la majorité des autres microprocesseurs (c'est-à-dire des processeurs tenant sur une seule « puce ») pour s'intégrer aisément dans les architectures parallèles très variées, grâce notamment à leurs quatre liens de communication. Outre le réseau maillé (B), on peut donc, par exemple, organiser les transputers en anneau (C) : dans ce cas, chaque transputer met à contribution au moins deux de ses liens bidirectionnels. Or cette architecture est simple mais lente, car un transputer peut être amené, pour communiquer avec un autre, à faire transiter son message par la moitié des transputers du réseau ! Pour pallier cet inconvénient, on peut donc utiliser un commutateur programmable, permettant de mettre en correspondance, à un moment donné, n'importe quel transputer de l'anneau avec n'importe quel autre (voir la figure 6). Dans ce cas, le transputer est exploité au mieux : ses quatre liens sont mis à contribution pour dialoguer directement avec n'importe lequel de ses homologues. Quant au réseau arborescent (D), qui exploite trois des quatre liens des transputers, il fait partie des architectures très bien adaptées à certains types d'applications, notamment en intelligence artificielle.

à exécuter plusieurs processus de façon quasi-parallèle. Son architecture le lui permet. En effet, le transputer peut partager son temps entre plusieurs processus. Ce « pseudo-parallélisme » est permis grâce à un gestionnaire de planification de tâches *(scheduler)* intégré dans le circuit, qui attribue à chaque processus le temps nécessaire à l'exécution de ses instructions entre deux rendez-vous. Certains processus peuvent même être « coupés en tranches » pour laisser la place à des tâches plus urgentes, si besoin est.

Dans une machine multiprocesseurs, le parallélisme peut être exploité de différentes manières (voir l'encadré). Le traitement en pipeline, qui autorise la décomposition d'une tâche en plusieurs étapes (sous-tâches) exécutées en cascade par autant d'unités de calcul, est aujourd'hui répandu — tous les supercalculateurs « classiques » font appel à cette technique, en décomposant les opérations à effectuer (additions, multiplications, etc.) en opérations élémentaires qui sont traitées à chaque étage de l'unité arithmétique et logique (voir « L'architective des nouveaux ordinateurs », dans ce numéro). Les traitements vectoriels (les processeurs sont organisés en vecteurs), ou matriciels (les processeurs sont organisés en matrices), sont beaucoup moins courants. Dans ce cas, les processeurs exécutent simultanément la même tâche sur des données différentes, les composantes des vecteurs ou matrices (on parlera d'*array processors)*. Une approche différente est celle qui est mise en œuvre dans le traitement systolique, caractérisé par la présence d'un ensemble de processeurs travaillant à la même cadence et échangeant des données au même moment. Le traitement concurrent enfin, ou asynchrone, voit un ensemble important de processeurs coopérer pour le traitement d'une application en parallèle, chacun traitant une tâche locale, à son rythme, en se synchronisant avec un autre quand ils échangent un message. Dans cette dernière classe, deux types d'architectures sont envisageables : à « connectique fixe » (les processeurs sont reliés entre eux par des liens physiques directs suivant une topologie donnée) et à « connectique reconfigurable » (le réseau d'interconnexion entre les processeurs peut varier dynamiquement en fonction de la structure du programme parallèle exécuté).

Parallélisme au pluriel.

Pour bâtir des machines parallèles à connectique fixe (on parlera également de réseau statique), la règle, nous l'avons dit, est de rechercher le compromis idéal permettant de minimiser le nombre de liaisons entre les processeurs sans pour autant trop augmenter la distance qui les sépare, autrement dit le nombre de processeurs intermédiaires à « traverser ». Partant de ce principe, les architectes de l'informatique ont imaginé de nombreuses topologies d'interconnexion des processeurs ou, plus précisément, des « nœuds » de la machine, chaque nœud pouvant comprendre un ou plusieurs processeurs, une mémoire locale et un composant de communication. Certaines sont plus spécialement adaptées à des applications particulières. C'est le cas de la structure arborescente (fig. 4), qui correspond bien à certains algorithmes utilisés en intelligence artificielle, ou du réseau maillé (les nœuds sont disposés en grille),

Figure 5. La société britannique Meiko a été l'une des premières à bâtir une machine, connue sous le nom de Computing Surface, faisant appel aux transputers (A). Elle a surtout été la première à commercialiser des modules (abritant en général chacun quatre transputers) qui peuvent être interconnectés entre eux, permettant ainsi la construction de machines parallèles modulaires. La société allemande Parsytec a repris la même idée pour l'architecture de son Megaframe Supercluster (B). Il comprend, dans sa version de base, 64 transputers. Des modèles plus puissants, affirme le constructeur, peuvent être obtenus en ajoutant des blocs supplémentaires de 64 transputers, sans limitation... (Clichés A, Meiko ; B, Parsytec)

bien adapté au traitement des images. D'autres sont plus universelles. L'architecture dite en « hypercube », généralisation du cube, apparaît aujourd'hui comme le compromis optimal (pour un coût moyen) des communications dans une topologie fixe. Mise en œuvre pour la première fois par C. Seitz et G. Fox au California Institute of Technology sous le nom de Cosmic Cube, cette topologie a été reprise par plusieurs constructeurs, dont, notamment, Intel Scientific Computers, division d'Intel Corp. (série iPSC), Thinking Machine Corp. pour sa Connection Machine (voir « L'architecture des nouveaux ordinateurs », dans ce numéro) et Floating Point Systems Corp. pour sa série T.

Les machines à connectique fixe.

Série T dans laquelle on retrouve le transputer qui n'est pas, ici, utilisé seul. Floating Point Systems, traditionnellement — comme son nom l'indique bien — fabricant de machines offrant de grandes performances de calcul en virgule flottante, a préféré développer lui-même le puissant processeur arithmétique dédié à ce type de calculs, qui voisine, dans chaque nœud, avec un transputer. Ce dernier, pour sa part, exécute les instructions simples, contrôle les échanges entre nœuds et le fonctionnement du processeur virgule flottante — les deux unités de calcul peuvent travailler concurremment pour atteindre, au plus, une puissance de calcul de 16 Mflops. Pour construire des machines de grande dimension, il fallait que chaque nœud puisse communiquer directement avec plus de quatre voisins « immédiats » : les quatre liens du transputer ont donc été multiplexés (c'est-à-dire partagés) pour en fournir seize. Quatorze de ces liens sont utilisés pour connecter chaque nœud à d'autres nœuds du système, deux liens sont utilisés pour l'intendance. Dans sa configuration minimale (un hypercube de dimension quatre), la série T comprend seize nœuds offrant une puissance théorique de 192 Mflops. Dans sa configuration maximale (un hypercube de dimension quatorze), la machine de Floating Point pourrait contenir 16 384 nœuds dotés chacun d'une mémoire (partagée par le processeur vectoriel et le transputer) d'un million d'octets, atteignant, au total, une puissance (très) théorique de 262 milliards d'opérations en virgule flottante par seconde — à titre de comparaison, signalons que la plus puissante des machines de Cray Research, numéro un mondial des supercalculateurs, possède une vitesse théorique de 2,5 milliards d'opérations en virgule flottante par seconde. Mais la performance de la machine de FPS n'a pas pu être évaluée, aucun calculateur de cette taille n'ayant encore été construit par la firme américaine...

Floating Point Systems, l'un des tout

premiers clients d'Inmos avec la firme britannique Meiko — d'ailleurs fondée par des anciens d'Inmos, pour mettre au point une machine connue sous le nom de Computing Surface (fig. 5) —, n'est plus depuis longtemps le seul à avoir inscrit sur son catalogue des produits faisant appel au transputer. A l'heure actuelle, près d'une centaine de constructeurs — européens pour la plupart — ne se contentent plus de développer des systèmes faisant appel à ce circuit, ils les vendent bel et bien.

Les réalisations et les projets en cours.

Leurs réalisations vont de la simple carte (destinée à s'insérer dans un micro-ordinateur — IBM-PC, Macintosh d'Apple, Atari-ST — ou une station de travail — Sun, Apollo — du marché, pour en augmenter les performances) à la machine spécialisée, en passant par les supercalculateurs. Nous ne pouvons les citer tous, mais notons la présence, parmi les pionniers, des firmes Inmos — on comprend aisément pourquoi — et Meiko. L'une propose des cartes comprenant quatre transputers interconnectés en anneau, qu'on peut assembler dans des machines de tailles plus importantes, appelées Item 400. L'autre fut la première à commercialiser une famille complète de modules permettant de construire des supercalculateurs flexibles et extensibles à base de transputers pour une large gamme d'applications et une clientèle prestigieuse qui s'étend des USA au Japon. Citons également Parsytec en République fédérale d'Allemagne (fig. 5), pour sa gamme de cartes fort complète pour « bus VME » et ses calculateurs Mega frame. Citons enfin des constructeurs d cartes spécialisées: Computer Syster Architects aux Etats-Unis et ses premier « accélérateurs » à quatre transputer destinés aux micro-ordinateurs Persona Computer d'IBM, Gemini Computer Sys tems en Grande-Bretagne pour ses carte graphiques comprenant jusqu'à neu T800, Niche Data Systems en Grande Bretagne pour ses cartes Sun, Transtec en Grande-Bretagne pour ses cartes gra phiques et contrôleurs fonctionnant su les micro-ordinateurs PC d'IBM, Perihe lion en Grande-Bretagne toujours pou ses cartes dédiées au micro-ordinateu Mega ST d'Atari, Sension et Quintek e Grande-Bretagne encore pour des carte destinées aux micro-ordinateurs PC d'IBM abritant jusqu'à dix-sept transputers Simtech en Norvège, pour ses cartes co processeurs, KKD au Japon pour son vi déophone assurant le traitement de images grâce à un réseau de six trans puters, Levco et Mechanical Intelligenc aux Etats-Unis ou Ergonomics en Suiss pour leurs systèmes destinés aux micro ordinateurs Mac SE et Mac II, IBM a Japon pour ses systèmes de reconnais sance d'images, Smith Associates e Grande-Bretagne pour ses ordinateur embarqués et ses machines de reconnais sance d'empreintes digitales... Citon également, pour la France, les produit logiciels et matériels présents ou à veni des sociétés d'avant-garde : Apsis (accé lérateur de simulation de circuits inté grés), Archipel (accélérateurs pour mi cro-ordinateurs et stations de travail) Telmat (cartes destinées à des machine appelées SM-90 et supercalculateur

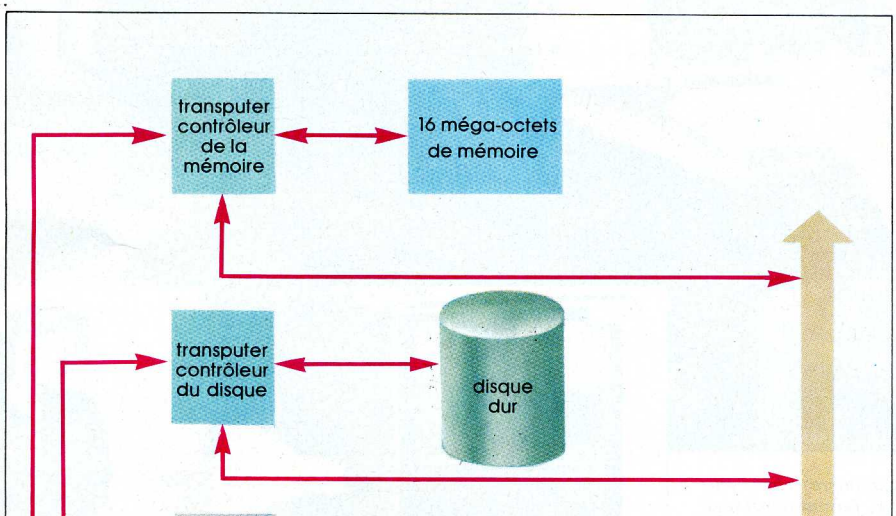
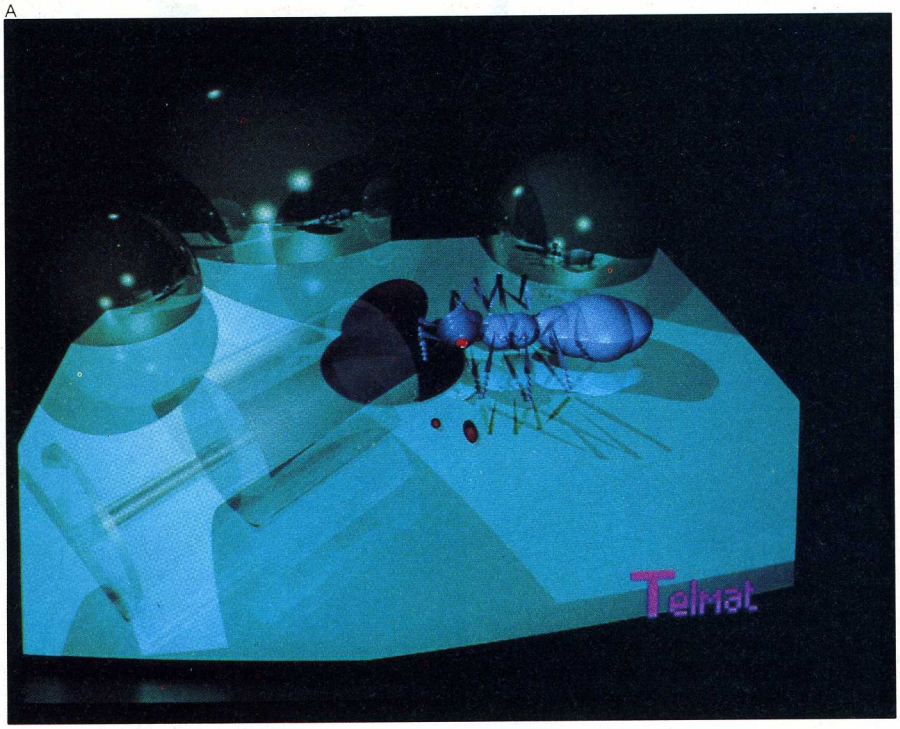


Figure 6. L'élément de base des machines super node (voir la figure 1) développées dans le cadr d'un projet Esprit réunissant Français et Britan niques est un « node ». Ce node comprend, dan la forme que nous avons schématisée ici, seiz transputers, chacun avec sa mémoire locale (jus qu'à 4Mo) dédiés au calcul (symbolisés sur c schéma par les carrés marqués d'une étoile) e trois transputers dédiés à des fonctions de gestio et de contrôle. L'un d'eux, notamment, (en bas droite sur le schéma) contrôle le bon fonctionne ment et la configuration de l'ensemble du nod grâce à une connexion particulière appelé « bus ». Tous ces éléments communiquent entr eux par l'intermédiaire d'un commutateur pro grammable (en bas, à gauche du schéma) auque ils sont reliés, qui joue un rôle similaire à celu d'un central téléphonique : à un instant donne n'importe quel transputer du réseau peut êtr connecté avec n'importe quel autre. L'arch tecture supernode dispose d'un atout majeur : ell est modulaire. Il est en effet possible d'inter connecter entre eux plusieurs nodes selon l même principe (il suffirait de remplacer, sur c schéma, chaque carré marqué d'une étoile pa un node entier). En outre, l'utilisation d commutateurs programmables pour relier le transputers présente un gros avantage : la ma chine n'est pas figée, son organisation intern peut prendre diverses formes topologiques, e fonction du traitement à effectuer. Du coup, ell sera toujours adaptée au mieux aux différent programmes de son utilisateur. Les reconfigura tions peuvent être réalisées statiquement (avan l'exécution d'un programme) ou dynamiquemen (le réseau est modifié en fonction des besoin pendant l'exécution d'un programme).

A



B

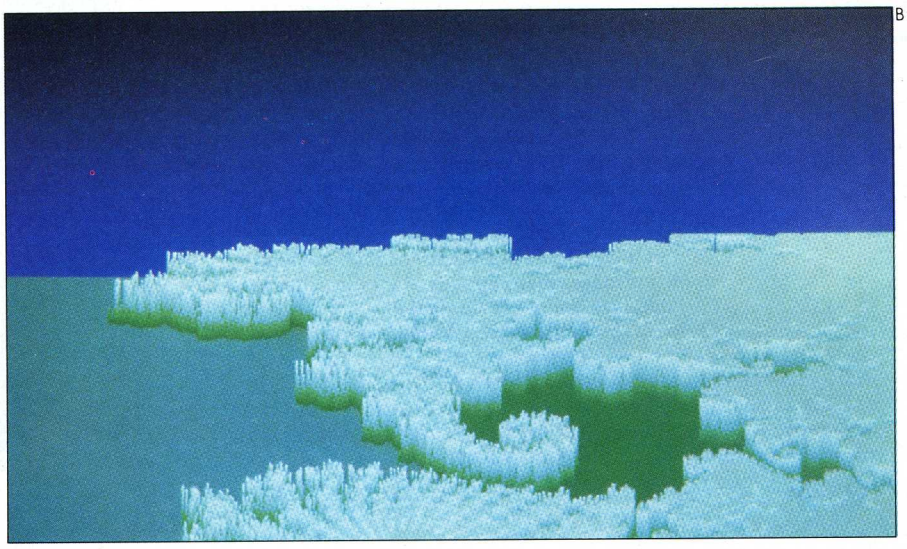


Figure 7. Ces images dites de synthèse ont été réalisées sur le calculateur T-Node, architecture de type supernode contenant 32 transputers, commercialisé par la société française Telmat. La première (A) a été obtenue en quelques minutes grâce à une méthode particulièrement gourmande en calculs mais permettant de restituer fidèlement les ombres portées, reflets et transparences, très nombreux ici. Par ailleurs, il a fallu seulement dix secondes au T-Node pour calculer l'image du célèbre « ensemble de Mandelbrot » (B) qui, bien que défini uniquement par des opérations mathématiques simples, est caractérisé par sa géométrie « fractale », extrêmement complexe. (Clichés Telmat)

T-Node issus du projet Esprit SuperNode sur lesquels nous reviendrons), CGA-HBS, filiale d'Alcatel NV (cartes pour machines de reconnaissance optique de caractères manuscrits et typographiques, destinées à des applications de tri postal) et CGEE, filiale de la Compagnie générale d'électricité (systèmes de contrôle industriels comprenant de 50 à 200 transputers).

En outre, de nombreux projets faisant appel aux transputers sont en cours, parfois à l'échelle européenne. Concurrent Supercomputer, mené à Edinburgh, utilise la Computing Surface de Meiko pour aboutir à un machine comprenant plus de mille transputers T800 et offrant des performances supérieures au milliard d'instructions en virgule flottante par seconde, dont les applications principales seront la simulation en mécanique des fluides, modèles biologiques, réseaux de neurones... Les projets Padmavati et Span pour leur part s'intéressent aux applications en intelligence artificielle. Ces projets auxquels participent Cimsa-Sintra et Thomson - CSF pour la France font partie du programme Esprit de la CEE. Parsifal, partie intégrante du programme britannique Alvey (l'équivalent, à l'échelle insulaire, du programme européen Esprit), est une machine groupant des modules appelés T-rack comprenant 64 transputers chacun, avec un à deux millions d'octets de mémoire par transputer. Son architecture est partiellement reconfigurable : un anneau réalisé avec deux des liens de chaque transputer reste figé alors que les deux autres liens sont configurables par programme. Alice est une architecture proposée elle aussi dans le cadre d'un projet Alvey, mené par l'Imperial College de Londres. La machine est prévue pour la fin de l'année 1989. Dans le même ordre d'idées, citons encore le projet Flagship, le plus important du programme Alvey (son coût approche les quinze millions de livres), réunissant le fabricant d'ordinateurs britannique ICL, l'université de Manchester et l'Imperial College de Londres.

Il n'est néanmoins pas certain que l'on pourra grandement accroître les performances des machines utilisant les transputers en les bâtissant sur des topologies fixes (ou partiellement reconfigurables) d'interconnexion. Pour obtenir les performances d'un supercalculateur — disons en moyenne 500 millions d'instructions en virgule flottante par seconde — il faudrait (dans un réseau purement statique) réunir quelque dix mille T414 ou environ un millier de T800. De plus, choisir une topologie fixe d'interconnexion signifie bâtir des systèmes plus ou moins bien adaptés à certains types d'applications. L'idéal serait, toutefois, d'obtenir des machines capables de traiter toutes sortes d'algorithmes avec la même efficacité. Pour obtenir ce résultat, mieux vaut ne pas relier directement les processeurs par des liens physiques inamovibles, mais par l'intermédiaire de commutateurs configurables par programme : tel est le principe du réseau reconfigurable (ou dynamique), dont la topologie se transforme au gré des applications, sans que l'utilisateur ait à intervenir. Nous rangerons dans cette catégorie le *crossbar* par exemple, une matrice de commutateurs programmables dont le principe est assimilable à celui des centraux téléphoniques : il permet d'interconnecter deux processeurs quelconques à un instant donné. Néanmoins, ce type de réseaux a la réputation d'être coûteux en matériel (le *crossbar* nécessite, pour connecter n processeurs, n^2 connexions). Le projet SuperNode, faisant appel à une architecture reconfigurable totalement originale, prouve qu'il est possible de construire une machine relevant de la même philosophie pour un prix moindre.

SuperNode, la machine reconfigurable.

SuperNode est un projet franco-britannique placé sous la bannière du programme Esprit, dont l'objectif est de construire une famille nouvelle de machines à connectique reconfigurable dynamiquement par programme, en utilisant les transputers T800 conçus par Inmos dans le cadre de ce projet. Démar-

suite page 1319

suite de la page 1315

ré en décembre 1985, il réunit Inmos, le Thorn-EMI, Royal Signals and Radar Establishment, l'université de Southampton pour la Grande-Bretagne et Apsis, Telmat et l'Imag (laboratoire de génie informatique de l'université de Grenoble) pour la France.

L'architecture supernode — choisissons délibérément les minuscules pour désigner et souligner le caractère générique et commun de ce type d'architecture, qui peut donner naissance à plusieurs classes de machines (T-Nodes fabriqués par Telmat en France ou SN1000 produits par Parsys, filiale de Thorn-EMI en Grande-Bretagne) — est une solution attractive à faible coût, une alternative aux topologies actuellement développées. Elle repose essentiellement sur la conception d'un module faisant appel à des transputers T800 (dont les liens peuvent être commutés par un composant spécial), une stratégie de construction hiérarchique et une méthodologie de programmation bâtie sur Occam. Cette approche devrait mener à l'élaboration d'une famille de supernodes allant du poste de travail individuel puissant au supercalculateur, en passant par des machines dédiées à une classe particulière d'applications. Le supernode est une architecture pouvant utiliser un nombre important de transputers tout en optimisant le coût des communications dans une machine de taille importante et en gardant un coût de fabrication très faible par rapport aux supercalculateurs existants de puissance équivalente.

DIFFÉRENTS TYPES D'EXPLOITATION DU PARALLÉLISME

Il existe essentiellement trois types d'exploitation du parallélisme. Le premier, dit *réplicatif*, signifie que chaque transputer, dans une « ferme » (ou atelier de processeurs), réplique et exécute le même code mais sur un ensemble différent de données. Cette forme de parallélisme indépendant (les processus communiquent peu) est assez triviale, d'une exploitation facile, mais largement utilisée pour la parallélisation d'applications existantes ou par les compilateurs (programmes de traduction du langage de haut niveau en langage directement compréhensible par la machine) de langages existants.

Le parallélisme dit *géométrique* ou « orienté structure des données » garde plus ou moins le même code exécutable pour chaque processeur ayant la responsabilité d'un sous-ensemble des données seulement. Par exemple, sur une matrice de taille importante, les opérations sur des sous-matrices sont appliquées localement aux données concernées. Le prix à payer pour l'optimisation géométrique des données dans une machine est un surplus de communication. En effet, les calculs locaux à une région accèdent à des données résultant des calculs effectués dans le voisinage ou sur des processeurs plus distants. Dans l'élaboration des algorithmes parallèles, le principe de localité est, dans ce cas, la règle d'or.

Enfin, le parallélisme *algorithmique* autorise son exploitation vraiment fine. Chaque transputer effectue uniquement une petite partie de l'algorithme — du programme — et les données traversent le réseau de transputer en transputer *(data flow)*. La communication entre processeurs doit être élaborée et efficace et la charge de travail correctement répartie. Le mélange de ces trois formes de parallélisme et son implantation dans une architecture parallèle de transputers au mieux adaptée aux besoins d'une application, permet d'obtenir les meilleures performances et l'utilisation d'un très haut degré de parallélisme physique.

Le supernode de base (que nous appellerons « node » pour éviter toute confusion) est typiquement constitué d'un ensemble de dix-huit transputers T800 (seize d'entre eux, dotés chacun d'une mémoire supplémentaire allant jusqu'à 4Mo, étant dédiés aux calcul, les deux autres assurant des fonctions d'intendance — accès à la mémoire de masse, sur laquelle sont stockées d'importantes quantités de données, ou contrôle du commutateur notamment). Ces briques de base sont interconnectées par un commutateur programmable qui n'est pas un simple crossbar (composé en réalité de plusieurs circuits développés à la demande des participants du projet par la firme japonaise NEC), appelons-le un configurateur, fonctionnellement semblable à un central téléphonique, qui assure en deux microsecondes la commutation des liens des transputers — n'importe quel lien d'un transputer peut, à un moment donné, être mis en relation avec n'importe quel lien d'un autre transputer (fig. 6). Notons au passage que cette configuration fonctionne aussi bien en mode MIMD que SIMD (voir « L'architecture des nouveaux ordinateurs », dans ce numéro). L'architecture d'un supernode est totalement modulaire : les nodes peuvent être interconnectés entre eux selon un principe similaire à la connexion des transputers décrite ci-dessus : le schéma précédent s'applique à nouveau, à la différence que les processeurs ne sont plus des transputers, mais des nodes de transputers. Dans ce cas, la machine est toujours reconfigurable : un commutateur électronique inter-nodes, sous le contrôle d'un transputer, permet de modifier la topologie entre ces nodes. Dans la configuration la plus étendue, avec le commutateur actuel, on reliera ainsi 64 nodes, c'est-à-dire 1 024 transputers dédiés au calcul.

Il est tout à fait possible de réaliser des topologies fixes (pipeline, arbres, hypercubes...) sur les supernodes, dont la connectique est contrôlée par programme — leur configuration peut même s'effectuer pendant l'exécution d'une tâche. Le langage initial de programmation est bâti sur Occam, mais d'autres types de langages sont en cours d'adaptation (C, Pascal, Fortran, Prolog, Ada, Lisp). Des programmes d'applications sont également développés par les divers partenaires du projet, en calcul scientifique, traitement du signal, imagerie, conception assistée par ordinateur, intelligence artificielle...

SuperNode n'est plus seulement un projet. Les sociétés Telmat et Thorn Emi, que nous avons déjà citées en tant que participants du programme, commercialisent d'ores et déjà des machines directement inspirées des développements réalisés dans le cadre du programme Esprit. Le calculateur de Telmat comprend, dans sa version de base, 16 ou 32 transputers T 800 (fig. 6 et 7). Organisés par groupes de huit (sur une carte), les transputers disposent chacun d'une mémoire locale plus ou moins importante. Le configurateur qui les relie permet d'obtenir toutes les topologies de connexions à l'intérieur du node, mais aussi de fournir à d'autres nodes toutes les combinaisons de liens souhaitées. Il réalise donc ce qu'on appelle un réseau « non bloquant », et cette propriété reste valable lorsque l'on organise hiérarchiquement des nodes entre eux, comme nous l'avons expliqué plus haut. Une carte contrôleur, équipée d'un transputer T 414, prend en charge la gestion d'un node et assure les communications avec le monde extérieur. Ce transputer contrôleur est ainsi responsable de son node. Mais si ce dernier est lui-même intégré dans un autre node, le contrôleur se soumettra à son homologue de l'étage supérieur. La reconfiguration du réseau peut être statique (pas de modification en cours d'exécution d'une tâche) ou dynamique (modification de la topologie au cours d'une tâche).

Les atouts des machines parallèles décrites dans cet article, et en particulier des machines reconfigurables, sont tels que l'on peut prédire, sans grand risque de se tromper, qu'elles deviendront rapidement populaires. Mais le chemin est encore long qui mène à la parfaite maîtrise de ces calculateurs et à leur exploitation maximale. Car la programmation parallèle n'est pas encore passée dans les mœurs des informaticiens. Un gros effort de développement — pour la mise au point d'outils d'aide à la programmation — et de formation reste indiscutablement à fournir. ■

Pour en savoir plus

- C.A.R. Hoare, *Communicating sequential processes,* Prentice Hall, 1985.
- Inmos Ltd, *Occam programming manual,* Prentice Hall, 1985 ; *Transputer reference manual,* 1986.
- G. Harp, C. Jesshope, T. Muntean, C. Whitby-Strevens, *Supernode : development and application of a low cost high performance multi-processor machine, Esprit 86,* Elsevier, 1987.
- T. Muntean (ed.), « Parallel Programming of Transputer Based Machines », Grenoble 1987, IOS-North Holland, 1988.
- T. Muntean, « SuperNode : architecture reconfigurable de transputers », *Revue BIGRE,* n° 56, 1987.
- A.W. Roscoe, C.A.R. Hoare, *The laws of Occam programming,* Oxford, 1986.
- R.W. Hockney, C. Jesshope, *Parallel computers-2,* Adam Hilger, 1988.