Entre la profecía de Moore y la ley de Amdahl

Julio Ortega Lopera
Departamento de Arquitectura y Tecnología de Computadores

Conferencia inaugural del curso 2008-2009 en la Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

A Carmen, Julio, y Diego

Introducción

Recientemente se ha medido una velocidad de cómputo de mil billones (10^{15}) de operaciones por segundo en un computador con alrededor de 120.000 procesadores. El dato se recoge en la lista TOP500 de Junio de 2008. Se ha alcanzado un objetivo que, si bien no estaba asociado a ninguna barrera física, tiene un efecto psicológico importante. Desde 1946, año en que entró en funcionamiento el ENIAC, se ha producido un factor de incremento en la velocidad de procesamiento de alrededor de un billón (10^{12}). Esto supone un ritmo de crecimiento exponencial de aproximadamente un 57% anual.

Efectivamente, uno de los lugares comunes cuando se habla de la evolución de los computadores es el rápido ritmo de mejora experimentado en todos los aspectos relacionados con el computador, desde la velocidad de los procesadores, hasta el tamaño y el tiempo de respuesta de la memoria, pasando por el espacio de almacenamiento disponible y el ancho de banda de comunicación que proporcionan las redes. Así, entre otros resultados, se ha ido reduciendo el tiempo de procesamiento, ha aumentado el volumen de datos que puede tratarse en tiempos preestablecidos, y han aparecido aplicaciones nuevas en las que los computadores encuentran utilidad.

En esta evolución de la capacidad de los computadores ha tenido bastante que ver la mejora de la tecnología de circuitos integrados, y

precisamente la ley de Moore, que establece un ritmo de crecimiento temporal de carácter exponencial para el número de transistores que se podrían incluir en un circuito integrado. Por otra parte, el aumento de las prestaciones de un computador tiene que ver con la identificación de sus cuellos de botella, tal y como se deduce de la ley de Amdahl que, además, plantea las dificultades del aprovechamiento eficiente del procesamiento paralelo. La innovación en ingeniería de computadores ha tenido lugar entre estas dos leyes, una empuja hacia la consecución de mejoras continuas en las prestaciones de las máquinas, y otra alerta respecto de la utilidad de las alternativas plausibles. De ahí el título de esta conferencia, que proporciona una aproximación a las fuerzas que, desde la perspectiva de la ingeniería de computadores, dirigen la evolución de los computadores. Para ello, expondremos nuestras respuestas a las siguientes preguntas:

(1) ¿Qué velocidad de procesamiento alcanzan los computadores actuales?,
(2) ¿Cómo se ha podido llegar a las prestaciones de los computadores actuales?,
(3) ¿Existen límites para la mejora de prestaciones?,
(4) ¿Cómo serán los computadores del futuro?.

Responder a las dos primeras preguntas *sólo* implica un trabajo de análisis de la situación actual y de lo que ha sucedido en el pasado. Sin embargo, las preguntas tercera y cuarta están relacionadas con el futuro. ¿En qué sentido puede llevarse a cabo una predicción de los resultados de la actividad innovadora que se desarrolla en una disciplina como la ingeniería de computadores?. Realmente ¿es eso posible?, ¿no es una contradicción en sí misma?. No hace falta más que recordar lo estrepitosamente erróneas que fueron afirmaciones como la de Thomas J. Watson, fundador de IBM, que en 1943 pensaba que sólo habría mercado mundial para unos cinco computadores, o la de Ken Olsen, de DEC, que en 1977 no veía ninguna razón por la que alguien podría querer un computador en casa. No obstante, parece que, aunque no sea posible prever detalladamente qué será lo nuevo, sí se pueden hacer ciertas

estimaciones sobre el nivel de prestaciones que pueden alcanzarse y, con ello, establecer los límites razonables para las propuestas de futuro. Y esto es lo que sí se puede enseñar y es en lo que el profesional debe mantenerse al día.

Según la definición de Michael J. Flynn ([FLY98]), la ingeniería es una profesión en la que se aplican principios científicos y matemáticos a las necesidades sociales. Una de las acepciones de la palabra proyecto en el diccionario de la Real Academia Española indica que proyecto es "conjunto de escritos, cálculos y dibujos que se hacen para dar idea de *cómo ha de ser* y lo que ha de costar una obra de arquitectura o de ingeniería". Si además tenemos en cuenta la definición de ingeniería de Flynn, plantear un proyecto viable implica poseer una imagen acertada del futuro, no sólo en el ámbito de su especialidad, sino también en los del entorno socio-económico al que se dirigen los resultados de su proyecto. Luego hay que completar las tareas y alcanzar los resultados que se han descrito en el proyecto. En cualquier caso, el trabajo del ingeniero se desarrolla a través de una sucesión de proyectos. Nuestras vidas profesionales y las de nuestros estudiantes transcurren entre proyectos. Elaboramos proyectos, gestionamos proyectos, completamos proyectos, evaluamos proyectos, etc.

Como se ha dicho, aquí se abordan cuestiones relativas a la innovación en el ámbito de la ingeniería de computadores. Además de proporcionar una visión de la posible evolución de los problemas que han de resolverse en el futuro y de las soluciones previsibles, llegaremos a algunas conclusiones respecto al *modus operandi* del ingeniero de computadores y a la forma en que se contempla la innovación en este campo. A pesar de que el destino de parte del trabajo que realizamos es la obsolescencia en menos de 18 o 24 meses (con suerte), hay conceptos, principios, e incluso artefactos que perduran más allá de los cambios y las mejoras tecnológicas, amén de haberlas hecho posibles. Ése es nuestro Grial como profesionales de la ingeniería en sus facetas docente e investigadora, la cima a la que Sísifo empuja su piedra para ver como vuelve a caer irremisiblemente.

La capacidad de los computadores actuales

Responder a la pregunta relativa a la capacidad máxima de los computadores actuales requiere algunas consideraciones. Existen diversos tipos de computadores que han surgido de especificaciones y requisitos muy diversos. Es conocida la clasificación que distingue entre computadores de uso personal, o computadores de sobremesa como resultado de la traducción del término inglés "*desktop computers*"; servidores; o computadores embebidos. En cada uno de esos grupos los objetivos de diseño son diferentes y, por tanto, pueden ser muy variadas las características relativas a la velocidad de procesamiento, prestaciones de la memoria, capacidad de almacenamiento, y necesidades de comunicación.

En cualquier caso, siempre podemos considerar los valores máximos de cada característica encontrados en alguna de las máquinas existentes. Así, si nos referimos a los límites en la velocidad de procesamiento, lo más razonable es acudir al TOP500 [TOP08], una lista que aparece dos veces al año (en Junio y Noviembre) desde 1993 e incluye los 500 computadores más rápidos instalados en el mundo. La rapidez se evalúa a partir de la velocidad de procesamiento del programa de prueba *Linpack*. Según la lista de junio de 2008, el computador más rápido era el *Roadrunner* de IBM que, al ejecutar el programa *Linpack*, alcanzaba una velocidad máxima superior a 1 PFLOPS, es decir, 1000 billones (10^{15}) de operaciones de números reales por segundo. Para dar una idea de lo que significan 1000 billones de operaciones basta con tener en cuenta que, si repitiésemos 1000 billones de veces algo que hacemos en un segundo, por ejemplo dar una palmada, tardaríamos casi 32 millones de años en dar esos 1000 billones de palmadas. ¡Cuántas vidas se podrían vivir en 32 millones de años!. Pues bien, imaginemos que esos 32 millones de años (¡todas esas vidas!) se concentran en un segundo. Además, el *Roadrunner* dispone de una memoria principal de 98 TBytes, es decir, 98 billones de bytes o lo que es igual, 784 billones de bits, distribuida entre más de 120.000 procesadores que funcionan a una frecuencia de 3.2 GHz. En 1946, el ENIAC, el

primer computador electrónico de propósito general realizaba 5.000 sumas por segundo, 357 multiplicaciones por segundo, ó 35 divisiones o raíces cuadradas por segundo, a una frecuencia de reloj de 100 KHz. Es decir, en algo más de 60 años se ha producido un factor de mejora de prestaciones de alrededor de un billón (10^{12}), con un reloj *sólo* 32.000 veces más rápido, pero 120.000 procesadores más trabajando en paralelo. Además, mientras que el ENIAC consumía unos 170 KW (aproximadamente 29×10^{-9} MFLOPS/W), el *Roadrunner* consume 2400 KW (437 MFLOPS/W), unas 14 veces más consumo, pero 15×10^{9} veces más eficiente energéticamente. A continuación analizaremos las condiciones que han hecho posible esta situación.

Los agentes de la evolución de los computadores

La ingeniería debe innovar. La innovación es el proceso que permite poner en el mercado una idea creativa. Una aproximación algo más detallada al concepto pone de manifiesto que la innovación implica creación, diseño, producción, uso y difusión de nuevos sistemas, productos, o procesos tecnológicos. Se deben proponer nuevas formas de superar las restricciones, de mejorar los rendimientos, de controlar la energía, etc. En esto se distingue de la ciencia que, en general, busca entender la realidad, no construir *artefactos* que la modifiquen. En ciencia, más que de innovación se puede hablar de creatividad, que se dirige hacia el desarrollo de nuevas herramientas de descripción, paradigmas, y procedimientos de resolución de ecuaciones. En la dinámica de la evolución científica no intervienen de forma relevante los intereses empresariales ni son tan determinantes los criterios de eficiencia económica. Sin embargo, en ingeniería hay que tomar bastantes decisiones determinadas por el ámbito de aplicación del diseño final, y la mayor o menor demanda (y los correspondientes beneficios) condicionan tanto el proceso de diseño en sí, como la tendencia a seguir en la disciplina [TRE95].

En cuanto al proceso de innovación tecnológica, existen modelos lineales que lo describen como una secuencia de etapas que van

desde la investigación al mercado, pasando sucesivamente por el desarrollo y la producción [KLI86]. Sin embargo, actualmente se está imponiendo la idea de que esos modelos lineales no son adecuados, sobre todo en campos donde el proceso innovador se desarrolla a gran velocidad, como es el caso de las industrias relacionadas con los semiconductores y las tecnologías de la información y las comunicaciones. Así, la innovación se contempla como un proceso emergente y dinámico, que implica una fuerte interacción entre redes complejas de tecnólogos e instituciones que se *autoorganizan* y *coevolucionan* junto con la tecnología [CLA88].

En este proceso evolutivo se distingue entre varios *patrones de innovación*. Los patrones de innovación que se proponen en [RYC99] son el patrón *normal*, el de *transición* y el de *transformación*. El patrón de *innovación normal* se caracteriza por una red estable de interacciones entre los distintos agentes que intervienen en los procesos tecnológicos y prácticamente ningún cambio en el proceso de diseño antes y después de la innovación. En cambio, los *patrones de transición y de transformación* se dan cuando hay innovaciones radicales o revolucionarias que dan lugar a nuevas redes y diseños tecnológicos. Si se producen modificaciones en la tecnología establecida se habla de *transición*, y si son cambios fundamentales hacia nuevos diseños se habla de *transformación*. Las nuevas redes que surgen con estos patrones pueden tener los mismos agentes participantes u otros que se van incorporando, pero siempre incluyen nuevos conocimientos, capacidades fundamentales, o ventajas complementarias [SCH04]. El conocimiento necesario para las *innovaciones transformadoras* puede no tener una relación clara con el conocimiento que una red o compañía ha alcanzado en el pasado y, por tanto requiere que la organización correspondiente sea bastante abierta en cuanto a la posibilidad de redefinir su estrategia, su estructura o sus procesos de toma de decisiones, y que sus miembros participen en otras redes, abarcando distintos sectores. Las *innovaciones de transición* modifican las tecnologías establecidas aprovechando el conocimiento que se ha ido acumulando a lo largo de la evolución bajo otros patrones de innovación, para superar restricciones que limitaban la mejora de prestaciones. En este caso

suelen surgir nuevas redes como resultado de la necesidad de cooperación, integrando las capacidades y conocimientos disponibles para superar los límites existentes. Cuando la evolución se produce según un patrón de *innovación normal* es cuando la tecnología puede proporcionar grandes beneficios por las innovaciones incrementales que los distintos competidores pueden llevar a cabo en paralelo. Las redes establecidas disponen del conocimiento necesario para dar respuesta a los problemas que surgen y las incertidumbres se refieren únicamente al tiempo y al coste del proceso necesario para solucionar el problema planteado.

La ingeniería de computadores aprovecha la tecnología electrónica de semiconductores, para la que el *patrón de transformación* se produjo en los años 40 y 50 con el descubrimiento y el desarrollo del transistor en los Laboratorios Bell. El transistor era una innovación basada en el conocimiento científico y dio lugar a una tecnología de diseño diferente de la que existía entonces. La posibilidad de una electrónica basada en componentes de estado sólido planteaba mejoras importantes en cuanto a prestaciones, fiabilidad, consumo, y tamaño, y no sólo supuso un cambio importante desde el punto de vista tecnológico sino que también hizo que emergiera una nueva red organizativa para explotar la nueva tecnología basada en el transistor. Por ejemplo, Texas Instruments, una compañía relacionada con los pozos de petróleo, fue la primera en fabricar un transistor de silicio.

La invención del circuito integrado y el desarrollo del *proceso planar* por parte de la empresa Fairchild (una empresa de suministros de aviación) corresponden a un *patrón de transición* y fueron decisivas en la evolución de la ingeniería de computadores. La conexión de componentes electrónicos discretos en un mismo sustrato, constituyendo los circuitos integrados, permite aumentar la funcionalidad de un circuito hasta un nivel determinado por la capacidad de integración. La situación en este caso es diferente de lo que ocurrió con la invención del transistor. Aquí se aprovechan tecnologías existentes y las organizaciones involucradas combinan sus conocimientos y habilidades a través de las redes de innovación

ya establecidas. Desde los años 60, la innovación en la tecnología de semiconductores ha seguido un *patrón normal*, con innovaciones incrementales hacia circuitos integrados CMOS cada vez con mayor densidad de transistores, más prestaciones y menores costes por unidad.

Así, la evolución de los computadores se ha producido gracias a un proceso de innovación tecnológica desarrollado por una comunidad de científicos y tecnólogos en centros de investigación, públicos y privados, en empresas y Universidades. Se ha llevado a cabo, por tanto, a través de un proceso distribuido, con una gran dosis de competitividad, pero también con los elementos de coordinación y cooperación necesarios para el desenvolvimiento de la ingeniería de computadores, y sin los cuales sería muy difícil entender las magnitudes de mejora resultantes. Concretamente, en ese proceso, la ley de Moore, que establece que el número de transistores en los circuitos integrados se dobla cada 18 o 24 meses, desempeña un papel central. Esta ley, que sobre cuyos límites nos detendremos más adelante, es el *paradigma de consenso* que marca el ritmo innovador en las empresas y demás agentes que intervienen en la tecnología de semiconductores y en la propia ingeniería de computadores. Como consecuencia de eso, ha tenido lugar un proceso de globalización y se han generado instrumentos como el ITRS (*Internacional Technology Roadmap for Semiconductors*) [ITRS], una hoja de ruta o *roadmap*, que ejerce el papel de mecanismo de coordinación entre los individuos de la colectividad de científicos y tecnólogos, permitiendo abordar la creciente complejidad que se produce en el ámbito de la tecnología de semiconductores. De hecho, un *roadmap* puede definirse como un cronograma de acciones o plan a seguir para alcanzar ciertos objetivos. Es una visión de futuro que incluye el conocimiento colectivo y la imaginación de los elementos más influyentes en el campo correspondiente. A través de ellos se atraen los recursos de las empresas y de la administración, se estimulan las investigaciones y se supervisa el progreso [GAL98]. Son los inventarios de posibilidades en cada campo.

A través del *roadmap* las empresas *tienen garantizado* que existirán proveedores que podrán proporcionarles la tecnología que requieren para mantenerse ellas mismas en lo que marca la hoja de ruta. En los 70, compañías grandes como IBM prácticamente eran capaces de fabricar casi todas las piezas de sus equipos, incluso diseñaban y construían sus propias salas limpias, incluyendo los sistemas más diversos que intervenían en las mismas, como por ejemplo el sistema de aire acondicionado. A medida que la industria madura, no se puede mantener esta forma de trabajo ya que se necesitan procesos más eficientes y se debe recurrir a terceros, especialistas en los distintos elementos del proceso de fabricación. Lógicamente, a las grandes empresas les interesa que sus proveedores sean económicamente viables, y eso sólo es posible si esos proveedores tienen una demanda suficiente. Para que eso sea posible, los grandes fabricantes deben promover que haya más empresas involucradas en la misma tecnología. A las empresas les interesa mantenerse en cabeza, pero no ser las únicas: *si eres el único no eres el líder, eres un solitario* [ISA00]. Y para estar en cabeza, hay que permanecer en movimiento, mejorando los procesos y manteniendo un nivel continuo de innovación. No basta con conocer una tecnología sino que hay que ser el mejor en su explotación.

Como se ha indicado más arriba, desde los 60, la industria relacionada con los computadores se caracteriza con un *patrón de innovación normal* en el que co-existen una gran cantidad de empresas y agentes entre los que están tanto los tecnólogos, como los investigadores de las Universidades que trabajan en los límites del conocimiento científico. Todos esos agentes se autoorganizan a través de estructuras como los consorcios de investigación, las alianzas y asociaciones estratégicas, las organizaciones para el desarrollo de estándares, e incluso a través de fusiones y adquisiciones de empresas. Además, aunque existen programas de investigación cooperativos a nivel nacional, como el programa de VLSI japonés de los 70 o la creación en los Estados Unidos de SRC y Sematech en los 80, se trata de un proceso global, como lo muestra el ITRS, que inicialmente fue una iniciativa de la SIA (*Semiconductor Industry Association*) americana y se denominaba

NTRS (*Nacional Technology Roadmap for Semiconductors*), hasta que en 1998 se incorporaron asociaciones de Europa y Asia.

Pero el ingeniero de computadores, en cualquier empresa o departamento, no sólo debe tener en cuenta los recursos que proporciona la tecnología de circuitos integrados, sino que también debe considerar los requisitos de las aplicaciones que demanda el mercado. Por lo tanto, para entender la evolución de los computadores hay que contar con la interacción de la *tecnología*, las *aplicaciones*, y los *mercados* con los principios de la *ingeniería de computadores*. Un ejemplo de la influencia mutua entre estos elementos se encuentra en el uso de redes de comunicación de altas prestaciones. Actualmente existen enlaces de red que proporcionan anchos de banda de varios *gigabits* por segundo. Para poder mantener esos anchos de banda, los servidores necesitan interfaces de red optimizadas, tanto en su hardware (buses y chipsets de altas prestaciones y tarjetas de red con prestaciones elevadas), como en el software de comunicación, y en la propia interacción entre hardware y software. Además, el procesador debería tener una velocidad de procesamiento elevada y casi no habría ciclos para las aplicaciones porque tendría que dedicarse casi por completo a las tareas de comunicación. Sin embargo, las aplicaciones determinan qué niveles de ancho de banda, dentro de los máximos posibles, son los realmente necesarios y, por tanto, qué arquitecturas de comunicación deberían implementarse según el mercado al que se dirige cada tipo de servidor.

Así, como muestra la Figura 1, por un lado, las mejoras de la tecnología electrónica han supuesto un aumento en la densidad de transistores y en la velocidad de conmutación, y están dando paso a *nuevas restricciones en el diseño de las microarquitecturas* en relación con el consumo de potencia y la comunicación local en el circuito integrado (el tiempo de una señal en cruzar un chip puede ser de decenas de ciclos de reloj). La consiguiente reducción en el costo por transistor, hace que las mejoras *tecnológicas impulsen nuevas aplicaciones y mercados* como el de los sistemas embebidos y la computación móvil. Por otra parte, esos *nuevos mercados generan*

presión sobre la tecnología exigiendo nuevas características de consumo de potencia, tamaño, y relación coste/prestaciones, y contribuyen a definir nuevas aplicaciones para las que hay una demanda potencial, determinando qué arquitecturas son las más aceptadas comercialmente. Por ejemplo, en la actualidad, parece que los servidores para el procesamiento de transacciones y los servidores Web dominan el mercado de servidores de gama alta, mientras que las aplicaciones multimedia y de procesamiento digital de señal mueven el de los computadores personales, portátiles y dispositivos móviles.

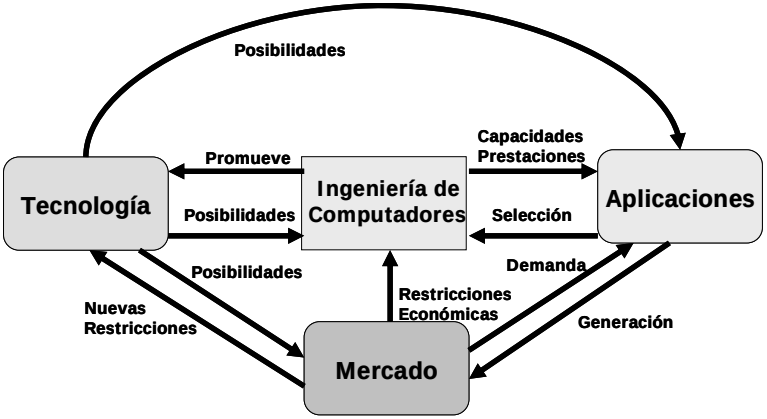

Figura 1. Interacciones en la evolución de los computadores [ORT05]

En cualquier caso, el *mercado impone al diseño de una arquitectura una serie de condicionantes* en cuanto a coste, prestaciones, consumo y tiempo de desarrollo. Un ejemplo de esta influencia social en la ingeniería de computadores está en las diferentes estrategias que siguieron IBM y CDC (Control Data Corporation) para el desarrollo de sus nuevos computadores en los años 60 [ELZ94]. En CDC, por un lado estaba la opción, liderada por Seymour Cray, de diseñar el computador más rápido posible en cada momento. Así, el CDC6600 (1964) se planteó con el objetivo de que fuera de 15 a 20 veces más rápido que su antecesor el CDC1604, y

para eso se recurrió a un repertorio de instrucciones más simple y a mejoras tecnológicas proporcionadas por Fairchild en cuanto a la velocidad de los transistores, de forma que era posible iniciar la ejecución de una instrucción cada 100 ns (algo impresionante en aquel momento). Posteriormente, en 1969, el CDC7600 cuadruplicaba las prestaciones del CDC6600 gracias a mejoras tecnológicas que permitieron utilizar un ciclo de reloj de 27.5 ns (frente a los 100 ns del CDC6600). Los computadores CDC6600/7600 eran completamente incompatibles con su predecesor, el CDC1604, y lo mismo ocurrió con su sucesor el Star-100, que fue el primer computador vectorial. En cambio, IBM no quiso sacrificar su penetración dentro del ámbito científico y comercial y, en el IBM 360, mantuvo su compatibilidad con los IBM 7090 y 7094, utilizando un repertorio de instrucciones más complejo que el CDC6600, que se consideraba más adecuado para las necesidades propias del ámbito comercial. Así, IBM mantuvo el dominio en el mercado del procesamiento de datos en el ámbito comercial (en el que CDC también intentó mantenerse con su CDC3600), mientras que CDC era el líder en el incipiente mercado de la *supercomputación*. En definitiva, las diferentes necesidades de las aplicaciones y condicionantes de los mercados determinaron la aparición de varias aproximaciones al diseño de computadores y con ello, la aparición de máquinas con distintos perfiles de prestaciones.

Finalmente, las *arquitecturas también afectan al desarrollo de la tecnología*, aunque en menor grado. Por ejemplo, la arquitectura von Neumann ha favorecido el desarrollo de una tecnología centrada en las mejoras en los procesadores y la memoria DRAM, en lugar de potenciar procesos tecnológicos que integren lógica en las memorias DRAM [VAJ01]. Por tanto, la influencia mutua entre tecnología, aplicaciones, mercados, e ingeniería de computadores tiene múltiples sentidos. Como muestra la Figura 1, se tienen ciclos de realimentación que complican la identificación de las causas primeras y los efectos finales, generando un comportamiento complejo, que dificulta la predicción de las tendencias futuras. Para entender los motores del cambio romperemos algunos de estos círculos de realimentación. Así, en primer lugar, nos detendremos en

las aplicaciones y los mercados, para pasar después a la influencia de la tecnología de circuitos integrados y su aprovechamiento a través de diversas innovaciones en ingeniería de computadores.

Las aplicaciones y los mercados en ingeniería de computadores

Inicialmente, el desarrollo y la fabricación de computadores estaban motivados fundamentalmente por aplicaciones de cálculo científico, y los computadores estaban financiados por entidades públicas y ligadas bien a intereses gubernamentales (militares en muchos casos), bien al entorno académico. Los usuarios del computador eran técnicos familiarizados con la aplicación a desarrollar, o en estrecho contacto con los especialistas en el problema a resolver. Esta situación ha cambiado a medida que las aplicaciones del computador y la accesibilidad al mismo han ido aumentando, implicando cada vez más a un mayor número de usuarios. La transformación del computador desde calculadora compleja a procesador de información, o a un elemento más dentro de los sistemas de telecomunicación, control, producción, etc. ha sido decisiva, no sólo en el desarrollo de la industria informática sino también en su incidencia económica y en la configuración de la denominada sociedad de la información.

Una consecuencia de la mencionada incidencia socio-económica del computador ha sido la creación de una industria informática al amparo de un mercado capaz de generar beneficios importantes. Esto ha hecho que en la evolución del computador hayan participado activamente las empresas y los condicionantes económicos. Si el computador se hubiese mantenido únicamente como un instrumento al servicio de la ciencia y la tecnología, la situación sería posiblemente distinta, quizá comparable a la de otros instrumentos científicos, como por ejemplo los radiotelescopios o los aceleradores de partículas. Existen empresas capaces de fabricarlos e instalarlos, pero normalmente son las entidades públicas las que los financian y su elevado coste hace que, para ser rentables, deban ser compartidos

por un elevado número de usuarios. Esta es, por otra parte, la situación usual en el ámbito de los grandes supercomputadores.

Muchas características de las arquitecturas de computador están determinadas por las denominadas *aplicaciones rompedoras* o, en una traducción más literal del término en inglés, las "*aplicaciones asesinas*" (*killer applications*). Se trata de aplicaciones que hacen que las plataformas y dispositivos en los que se implementan y permiten acceder a las mismas sean ampliamente demandados. Por ejemplo el correo electrónico (y sus derivados) puede considerarse una aplicación de este tipo, y ha contribuido de manera indudable a incrementar la demanda de acceso a los computadores en todos los ámbitos, incluyendo el doméstico. Las características de estas aplicaciones, y sus requisitos de cómputo condicionan el desarrollo de las nuevas arquitecturas de computador ya que el uso que una aplicación hace de los recursos de la arquitectura determina, tanto las prestaciones alcanzadas, como las características de la arquitectura cuya mejora es más conveniente para aumentar esas prestaciones. Según la ley de Amdahl [AMD67], la ganancia de velocidad que se consigue al mejorar un recurso de un computador en un factor igual a p está limitada por la cota $p/(1+f(p-1))$, donde f es la porción del tiempo en que dicha mejora no se utiliza en las aplicaciones que se ejecutan en el computador. De esta forma, por mucho que mejoremos un recurso (es decir, por mucho que incrementemos p), la mejora de velocidad no va a ser mayor que $1/f$. Si el recurso que mejoramos es un cuello de botella, f será pequeña y la ganancia de velocidad alcanzable puede ser elevada, pero si las aplicaciones utilizan poco ese recurso, f es próximo a 1, y la mejora en la que se ha invertido esfuerzo se desperdiciaría.

Por tanto, es importante tener en cuenta el perfil de las aplicaciones a las que se dirige un computador. De esta forma, se pueden identificar los elementos clave y las cuestiones recurrentes e invariantes en las arquitecturas de los computadores futuros. No se pretende aquí hacer un estudio exhaustivo de todas las aplicaciones que han tenido interés desde distintos puntos de vista (económico, científico, etc.) sino hacer una breve referencia a algunas de las que

han influido en el desarrollo de nuevas arquitecturas (y pueden verse influidas por ellas, según ilustra la Figura 1) y, con bastante certeza, van a seguir haciéndolo en los próximos años. Así, nos referiremos a las aplicaciones de *cálculo científico* que precisan del uso de supercomputadores, la *computación ubicua* (*pervasive computing*) y las aplicaciones de *internet*, y las *aplicaciones multimedia*.

En primer lugar, se pueden considerar las aplicaciones de cálculo científico que demandan prestaciones elevadas, y para las que distintos programas de investigación financiados normalmente por gobiernos u organizaciones públicas han permitido que se mantenga una cierta industria de supercomputadores, cuyos resultados han ido pasando a arquitecturas comerciales y a los microprocesadores, a medida que la tecnología lo ha hecho posible. Así, en 1992, el Departamento de Defensa los Estados Unidos de Norteamérica planteó el *High-Performance Computing and Communication Program (HPCC)* y se identificaron una serie de *aplicaciones-reto* (*Grand Challenge*) que desbordaban las capacidades de cómputo disponibles. Como se ha indicado, desde 1993, el TOP500 [TOP08] permite seguir la evolución de la tecnología y las prestaciones de los supercomputadores. Es posible citar un número considerable de aplicaciones que, junto a las aplicaciones-reto del HPCC, necesitan recursos de cómputo elevados, tanto en cuanto a capacidad de procesamiento, como a memoria, almacenamiento y comunicaciones. Así tenemos el diseño de medicamentos, la simulación de los océanos y del clima a escala global, el diseño de estructuras de proteínas, la predicción del tiempo cada vez a más largo plazo, la determinación del genoma humano, las simulaciones de dinámica molecular, el modelado de semiconductores y superconductores, etc.

Para dar una idea de las necesidades de estas aplicaciones podemos tomar como ejemplo la *simulación de biomoléculas* [NGO92]. A través de ella se alcanza conocimiento acerca de los procesos biológicos en los que intervienen (proteínas, DNA, RNA, encimas,..), complementando el trabajo de la biología experimental. La estructura, dinámica, y función de los complejos biomoleculares implica fenómenos que se desarrollan en escalas de tiempo y espacio

bastante diferentes: tiempos entre 10^{-15} y 10^{-3} segundos (12 órdenes de magnitud) y dimensiones entre 10^{-10} y 10^{-8} metros (2 órdenes de magnitud). Actualmente se pueden simular periodos de tiempo del orden de 10^{-8} s como mucho, cuando se debería llegar a periodos de 10^{-3} s. Así, en el problema del plegamiento de proteínas, si se considera una proteína típica en disolución acuosa, con unos 32.000 átomos y una duración para el proceso de plegado de unos 100 microsegundos (10^{-4} s). El número de pasos de simulación necesarios es del orden de 10^{11} si consideramos que cada paso de simulación corresponde a unos 10^{-15} segundos del proceso. En cada paso de simulación hay que procesar un número aproximado de 10^9 interacciones en el cálculo de fuerzas y el procesamiento de cada una de esas interacciones implica 10^3 instrucciones. Por tanto, el número total de instrucciones máquina sería de 10^{23} ($10^{11}\times10^9\times10^3$) y, suponiendo una velocidad de procesamiento de 10^{15} instrucciones/s (es decir un supercomputador con prestaciones del orden de los petaflops), se necesitarían más de tres años de simulación ($10^{23}/10^{15}=10^8$ s). Junto con las aplicaciones intensivas en computación, tampoco hay que olvidar las que necesitan procesar volúmenes elevados de datos con una complejidad computacional cada vez más elevada. Ejemplos de estos *problemas intensivos en datos* se encuentran en el tratamiento de datos recogidos por satélites en aplicaciones de astronomía o geofísica [XUE08]; la anticipación, detección y respuesta a ataques en la red; etc. Mientras que actualmente se empiezan a abordar problemas con conjuntos de datos del orden del petabyte y secuencias de datos del orden de los gigabits, en unos años, la frontera pasará a situarse en los exabytes [GOR08].

Una de las consecuencias de la ley de Moore es el rápido ritmo de abaratamiento de la capacidad de procesamiento. Así durante la pasada década, se estima que el costo de la *capacidad* de cómputo se ha reducido en un factor de entre el 30 y el 100%, con lo que el número de usuarios que tienen accesibilidad a una capacidad de procesamiento dada crece exponencialmente, afectando a la rentabilidad de las aplicaciones. Por lo tanto, además de utilizarse en video-juegos, lectores de DVD, teléfonos móviles, impresoras,

escáneres, televisores, etc., los computadores embebidos están sustituyendo a los circuitos electrónicos analógicos en multitud de aparatos y dispositivos, que pueden mejorar así sus prestaciones, fiabilidad, y funcionalidad, de una forma inalcanzable sólo a través de mejoras en la circuitería analógica. Estas aplicaciones han sido posibles gracias a la capacidad tecnológica y de diseño, que ha permitido el desarrollo de los computadores embebidos y sistemas en un chip (SoC), que por otra parte se ven impulsados por el mercado que existe para todas las aplicaciones en las que son pieza clave. Así, se habla de *ubiquitous computing* o *pervasive computing* (términos usualmente traducidos por computación ubicua) para referirse al hardware, al software y a las aplicaciones relacionadas con la computación móvil, los computadores de bolsillo y embebidos, la interacción hombre-máquina y máquina-mundo real, etc. En este escenario que plantea la computación ubicua, la comunicación entre dispositivos de cómputo es fundamental. De hecho internet y la *web* son, por tanto, los elementos clave ya que la conexión de los equipos a la red ha llegado a ser casi más importante que cualquier dispositivo de cómputo [HEN03]. Los servicios son las *aplicaciones rompedoras* en este mundo de computadores interconectados y recursos distribuidos, planteándose una gran demanda de *servidores* que ejecuten aplicaciones a través de la red con un adecuado nivel de prestaciones, que no se refieren sólo a la velocidad, sino también a la disponibilidad, facilidad de mantenimiento, seguridad, y escalabilidad de los servicios proporcionados.

El desarrollo de Internet ha hecho que adquieran relevancia aplicaciones como el comercio electrónico, la enseñanza a distancia, la televisión digital e interactiva, la publicidad dirigida, etc. Muchas de esas aplicaciones utilizan en mayor o menor medida el procesamiento conjunto de vídeo y audio continuos. Este *procesamiento media* o *multimedia* está caracterizado por los grandes volúmenes de datos a tratar (aunque estén comprimidos), por las restricciones de tiempo real existentes para la recepción conjunta de las señales, y por su continuidad en el tiempo. Esta situación plantea la necesidad de disponer de sistemas capaces de acceder, almacenar, procesar, y distribuir estos ficheros *media* a través de entornos de

comunicación heterogéneos. Para hacer posible esto, además de la conexión a través de Internet y nuevos procedimientos de compresión para reducir el tamaño de los datos sin afectar a la calidad, sería necesario disponer de tecnologías avanzadas que permitan almacenar los elevados volúmenes de datos que se utilizan, y de la capacidad de procesamiento suficiente en el computador donde se ejecuta el cliente. Esta necesidad de ampliar la capacidad de procesamiento de aplicaciones multimedia ha influido en la incorporación de repertorios de instrucciones que operan con vectores (o repertorios multimedia) no sólo en procesadores de propósito específico sino también en procesadores de propósito general, como es el caso de las extensiones MAX (1993) en HP, VIS (1995) en Sun, MVI (1997) en DEC, MMX (1997) y SSE (1998) en Intel, y 3DNow! (1998) en AMD, para aprovechar el paralelismo de datos típico en estas aplicaciones.

La tecnología de circuitos integrados en ingeniería de computadores

Por otro lado está la influencia de la tecnología en la ingeniería de computadores. La tecnología de integración ha permitido reducir el tamaño de los transistores y aumentar la superficie de los circuitos integrados. Por un lado, estos dos efectos actúan favorablemente en el mismo sentido de aumentar el número de transistores. Por otro, la reducción de las dimensiones del transistor ha incrementado su velocidad de conmutación. Al disponerse de más transistores cada vez más rápidos, la capacidad de cómputo del circuito integrado crece. Ahora hace falta utilizar eficientemente estos recursos.

La estrategia que ha seguido la arquitectura de computadores para aprovechar las mejoras tecnológicas que proporcionaban cada vez más transistores en un chip se puede resumir en dos principios: *paralelismo* y *localidad*. El paralelismo se ha aprovechado a través de la segmentación de cauce y de la repetición de elementos (la utilización de varios procesadores iguales, o con funciones

diferenciadas como es el caso de los controladores de DMA, los procesadores de E/S, etc.). La localidad se ha manifestado en las jerarquías de memoria. No obstante, tanto para implementar el paralelismo como para mejorar la localidad hay que utilizar recursos, sobre cuya distribución debe decidir el diseñador, en base a la comprensión de los efectos posibles en las prestaciones del sistema. Esta interesante interacción paralelismo-localidad se da en el computador a todas las escalas, tiene efectos en los elementos y en los problemas que se plantean en el computador (por ejemplo la coherencia de *cache*, o la necesidad de estructuras de comunicación entre procesadores, procesador y memoria, etc.), y permite exponer los conceptos esenciales que se plantean al diseñar/evaluar/utilizar un computador.

Es posible aproximarse a la influencia de la tecnología y a las estrategias de la ingeniería de computadores a partir de un modelo frecuentemente utilizado para evaluar las prestaciones del procesamiento de una tarea en un computador, un modelo sencillo pero suficiente para nuestro propósito:

$$T_{tarea} = NIxCPIxT_{ciclo} = NIx\left(\frac{CPI}{f}\right)$$

donde T_{tarea} es el tiempo que tarda en ejecutarse un programa determinado, *NI* es el número de instrucciones máquina de dicho programa que se ejecutan, *CPI* es el número medio de ciclos por instrucción, y T_{ciclo} el periodo de reloj del procesador (inverso de la frecuencia, *f*). La ingeniería de computadores abarca los aspectos que determinan los valores de *NI*, *CPI*, y *f*, y su interrelación. Así, *NI* depende del *repertorio de instrucciones* y del *compilador*; *CPI* viene determinado por el *repertorio de instrucciones* y la *organización del computador*; y *f* depende de las prestaciones que proporcione la *tecnología* y de la *organización del computador*. Como se ha indicado, la organización del computador afecta tanto al valor de *CPI* como a *f*, ocasionando que realmente la métrica importante sea el cociente *CPI/f*. Aunque se utilicen frecuencias mayores, si se produce un incremento del mismo orden en *CPI*, las prestaciones no aumentarían, en cambio,

se conseguirían mejoras importantes de prestaciones si aumenta *f* y disminuye *CPI* al mismo tiempo.

El paralelismo implícito en una aplicación puede ser de diferente tipo (paralelismo de tareas y paralelismo de datos) y se puede hacer explícito en diferentes niveles (*paralelismo entre instrucciones, paralelismo entre hebras, entre procesos*, etc.) para que lo pueda aprovechar la arquitectura. El paralelismo entre instrucciones ILP (*Instruction Level Parallelism*) se ha aprovechado a través de *procesadores segmentados* en los que se han introducido mejoras como la posibilidad de enviar a ejecutar varias instrucciones al mismo tiempo, la ejecución desordenada, el procesamiento especulativo, etc. Se puede reflejar esta situación en la expresión del tiempo de ejecución, si se introduce en ella el número medio de ciclos entre emisiones de instrucciones, *CPE* (Ciclos entre Emisiones), y el número medio de instrucciones que se emiten cada vez, *IPE*. Así, el valor de *CPI* se puede expresar como *CPE/IPE*, y el tiempo como:

$$T_{tarea} = NI \times \left(\frac{CPE}{IPE} \right) \times T_{ciclo} = NI \times \left(\frac{CPE}{IPE} \right) \times \left(\frac{1}{f} \right)$$

En un procesador segmentado (no superescalar), $CPE = IPE = 1$, y $CPI = 1$; en uno superescalar debería ocurrir que $CPE = 1$ e $IPE > 1$ (en cada ciclo se emiten varias instrucciones); y en un procesador no segmentado se tendría que $IPE=1$ y $CPI = CPE > 1$ (se emite una instrucción cada ciclo máquina constituido por varios ciclos de reloj).

La ingeniería de computadores ha aprovechado las mejoras en la tecnología de integración a través del diseño de microarquitecturas cada vez más complejas que persiguen reducir el número medio de ciclos por instrucción procesando instrucciones en paralelo, y funcionan a frecuencias cada vez mayores. El tamaño de los dados de silicio en los que se implementan los circuitos integrados es cada vez mayor, de forma que un incremento en un factor *b* en el lado del circuito integrado supone un incremento en un factor b^2 en la superficie disponible para transistores. Al mismo tiempo, el tamaño de los transistores se ha ido reduciendo y ha crecido el número de capas

para interconexiones, de forma que una reducción en un factor *a* en el tamaño del transistor implica aproximadamente un aumento del mismo orden en su velocidad de conmutación, y un aumento de un factor a^2 en la densidad de transistores en el circuito integrado. Cuanto mas rápidos sean los circuitos, a mayor frecuencia puede funcionar el procesador (mayor es *f*), y cuantos más transistores se tengan en un circuito integrado más complejas pueden ser las microarquitecturas del procesador, para permitir la ejecución paralela eficiente de más instrucciones. De esta forma se reduce el número medio de ciclos por instrucción, *CPI*, aumentando las instrucciones que se emiten, *IPE* y/o reduciendo los ciclos entre emisiones, *CPE*.

En la década pasada, los microprocesadores han estado mejorando sus prestaciones en torno al 50–60% anual (aproximadamente se han doblado cada 18 meses), tal y como la ley de Moore indica para el número de transistores en un circuito integrado. Como se ha indicado, este ritmo se ha debido al aprovechamiento de las mejoras tecnológicas para *aumentar el número de instrucciones procesadas por ciclo (IPC), y la frecuencia de reloj de los procesadores*. El aumento del número de instrucciones por ciclo en la década de los 90 se ha conseguido gracias a innovaciones muy relevantes en las microarquitecturas de los procesadores.

Entre estas innovaciones se pueden destacar la *ejecución desordenada* de instrucciones y el *procesamiento especulativo*, que describimos a continuación brevemente. El procesamiento paralelo de las instrucciones se ve limitado por las dependencias entre las propias instrucciones y los datos que procesan. Por ejemplo, en la secuencia de instrucciones que se muestra en la Figura 2, en la que suponemos que los registros *r1*, *r2* y *r3* tienen valores válidos que se han asignado anteriormente, la instrucción (2) necesita el dato que proporciona la instrucción (1), y no podría empezar a ejecutarse antes de que se haya completado el acceso a memoria que lo proporciona. Todas las instrucciones tendrían que esperar a que terminase la instrucción (1). No obstante, la instrucción (4) no depende ni de la instrucción (1) ni de la (2). Tan sólo existe un

problema: la instrucción (2) y la instrucción (4) escriben sus resultados en el mismo registro, *r5*.

```
(1)     load r4,8(r1)    ; cargar r4 (r1 apunta al dato)
(2)     mult r5,r2,r4    ; obtener r5=r2*r4
(3)     add r6,r5,r3     ; obtener r6=r5+r3
(4)     add r5, r1,r2    ; obtener r5=r1+r2
(5)     bz r5 etiq       ; saltar a etiq si r5 es cero
(6)     load r7, (r5)    ; cargar r7 (r5 apunta al dato)
(7)  etiq: ….
```

Figura 2. Ejemplo de secuencia de instrucciones máquina

Los procesadores superescalares incluyen el hardware necesario (en las denominadas *estaciones de reserva*) para detectar las instrucciones que tienen disponibles tanto sus operandos como el circuito donde se tienen que ejecutar, y las emiten para que empiece su ejecución tan pronto como sea posible. Al cambiar el orden en que se ejecutan las operaciones hay que tener cuidado con el orden en que escriben sus resultados (el orden en que las instrucciones (2) y (4) escriben en *r5*). Para ello se aplica la *técnica de renombramiento* de registros que asigna ciertos registros internos del procesador (que usualmente constituyen el denominado *buffer de renombramiento*) para que las instrucciones escriban temporalmente sus resultados, y disponen de estructuras como el *buffer de reordenamiento* (ROB) para asegurar que los resultados que se encuentran en esos registros internos se escriban en los registros de destino indicados por las instrucciones, en el orden en que aparecen en el programa (que la instrucción (2) escriba primero en *r5* y después la instrucción (4)) a pesar de que las operaciones se hayan podido ejecutar en otro orden.

Por otra parte, los procesadores actuales son grandes especuladores. Cuando llega una instrucción de salto que depende de una condición que todavía no se ha terminado de evaluar, el cauce debería detenerse dado que no se conocería si producirá el salto o no. Para evitar el descenso de rendimiento que supondría esta detención, los procesadores utilizan información de las anteriores ejecuciones de la

instrucción de salto o del comportamiento general de dichas instrucciones para determinar la opción más probable, y continúan procesando instrucciones según dicha información. Así, en la secuencia de instrucciones anterior, si cuando llega al procesador la instrucción de salto (5), la instrucción (4) no se ha terminado de procesar, el procesador predice si es más probable que se produzca el salto a la dirección *etiq* o que se continúe con la instrucción (6). El procesador dispone de recursos para realizar esta predicción, incluyendo memoria para almacenar la historia pasada de las instrucciones de salto, y recursos para recuperar la secuencia de instrucciones correcta en el caso de que no se haya acertado en la predicción. La eficacia del procedimiento de salto junto con los requisitos de memoria para guardar los datos de la historia de las instrucciones son aspectos fundamentales en los microprocesadores actuales.

En general, tanto las frecuencias de reloj como el valor de *IPC* de los microprocesadores han ido creciendo. Sin embargo, con el incremento en la densidad de integración, estos dos factores de mejora pueden ver limitados sus ritmos de crecimiento debido a la mayor influencia de aspectos que se pondrán de manifiesto con más intensidad a medida que se vaya pasando desde procesos de 130 nm (año 2002) a procesos de 32 nm (previstos para el año 2010). Así, ya a comienzos del siglo XXI se contemplaba que se podría pasar de ritmos de crecimiento de prestaciones del 50% anual, a mejoras de entre un 12% y un 17% [AGA00] si se mantenía la misma estrategia en el diseño de los microprocesadores. Concretamente, a medida que el tamaño característico de la tecnología de integración se reduce, los retardos de transmisión se hacen mayores que los tiempos de conmutación de los transistores y, con ello, el número de transistores accesibles por ciclo para la señal correspondiente. Así, el retardo de las líneas más largas del chip disminuiría a un ritmo de dos a cuatro veces menor que el retardo de las puertas lógicas. Esto es debido a que, al reducirse las dimensiones en un circuito integrado, disminuyen la anchura y la altura de las líneas, presentando una menor sección y por tanto mayor resistencia. Por otra parte, la capacidad de las líneas no disminuye en la misma medida que aumenta la resistencia, dado que

aunque la superficie del conductor es menor, también lo es la distancia entre líneas de una misma capa. De esta forma, la disminución en la capacidad asociada a la superficie del conductor, se ve contrarrestada por el incremento en la capacidad de acoplamiento entre líneas distintas [AGA00]. Al tiempo que las líneas se hacen más lentas, los transistores se hacen mucho más rápidos al ser el tiempo de conmutación, en primer orden, proporcional a la longitud de puerta.

Otra consecuencia del aumento de la densidad de transistores y del funcionamiento a frecuencias mayores es la elevada potencia disipada por los circuitos integrados. Esta situación está motivando que la potencia sea un factor importante también en el diseño de microprocesadores de propósito general: en los procesadores embebidos, como es lógico dadas las características de los productos donde se incluyen, las restricciones relacionadas con el consumo de energía y la disipación de calor ya eran esenciales. Muchas veces se olvida el aspecto relacionado con el consumo energético de los equipos informáticos, pero en un artículo del año 2000 del *Financial Times* se indicaba que el consumo de potencia atribuible a las tecnologías de la información representaba un 8% del consumo total en los Estados Unidos de Norteamérica. El *Roadrunner* de IBM con más de 120.000 procesadores consume unos 2.4 Megavatios.

Para entender la influencia de la tecnología en la potencia disipada en un circuito integrado se puede utilizar la expresión [MUD01]

$$Potencia = ACV^2 f + tAVI_{corto} + VI_{leak}$$

donde el primer sumando corresponde al consumo dinámico al cargar y descargar la capacidad a la salida de una puerta (C), el coeficiente de actividad del circuito (A) relacionado con la proporción de transistores que conmutan en cada ciclo; la tensión de alimentación (V); y la frecuencia de reloj (f). El segundo sumando se debe a la corriente (I_{corto}) que fluye entre fuente y tierra durante un instante (t) al conmutar la puerta. Por último, el tercer sumando es la potencia consumida debido a la corriente de pérdidas (I_{leak})

independiente del estado de la puerta. Es posible reducir la potencia consumida disminuyendo la frecuencia de funcionamiento (a costa de una reducción en las prestaciones), o disminuyendo la tensión de alimentación. No obstante si se tiene en cuenta que la frecuencia máxima a la que puede funcionar el circuito es

$$f_{\max} \propto \frac{(V - V_{umbral})^2}{V}$$

habría que reducir la tensión umbral (V_{umbral}) para no reducir las prestaciones, pero esto supondría aumentar exponencialmente el valor de la corriente I_{leak}, ya que

$$I_{leak} \propto e^{\frac{-qV_{umbral}}{KT}}$$

Por lo tanto, aunque la reducción de la tensión de alimentación implique la reducción del consumo sin afectar a las prestaciones, el aumento de consumo energético asociado a la corriente de pérdidas limita el alcance de esta técnica (más allá de un valor mínimo de tensión). Así, se están investigando técnicas basadas no sólo en el diseño del circuito o la microarquitectura (que tendería también hacia la simplicidad y a la reducción del coeficiente de actividad), sino que también consideran los niveles de arquitectura y de sistema operativo. Por ejemplo, el sistema operativo supervisa la operación del procesador para ajustar su frecuencia de funcionamiento a distintos niveles de compromiso entre las prestaciones requeridas por las aplicaciones y el consumo. Al nivel de la arquitectura, se intenta aprovechar el paralelismo (para reducir la frecuencia del procesador) y optimizar el uso del sistema de memoria y los buses [MUD01]. Todo esto pone de manifiesto la interacción entre los distintos niveles del computador.

En los últimos años, el problema de los retardos, y la necesidad de limitar los consumos de energía han promovido procesadores en los que un mayor aprovechamiento del paralelismo entre instrucciones no

se alcanzase a través de una microarquitectura con mayor complejidad hardware, sino que es el compilador el que asume la responsabilidad de conseguir una planificación de instrucciones óptima. Se trata de los procesadores VLIW (*Very Long Instruction World*), del que el procesador Itanium de Intel es un ejemplo.

La otra opción la constituyen los denominados multiprocesadores en un chip (CMP) o microprocesadores multi-núcleo o *multi-core*, con varios procesadores dentro del circuito integrado que constituye el microprocesador. IBM fue la primera compañía que, en el año 2000, lanzó al mercado un microprocesador con dos procesadores. Se trataba del POWER4, que estaba orientado al mercado de los servidores e incluía dos procesadores POWER (*Performance Optimization With Enhanced RISC*) a 1 GHz, memoria cache de segundo nivel (L2) y las marcas de la cache del nivel tercero (L3). El POWER4 fue seguido por otros dos microprocesadores duales, el POWER5 en 2004, con dos procesadores multi-hebra simultánea (SMT) de frecuencias entre 1.4 y 2 GHz, y el POWER6 en 2007, entre 3.5 y 4.7 GHz, y tecnología de 65 nm. En el segmento de los procesadores de sobremesa el primer microprocesador multi-núcleo fue el Pentium D serie 800 de Intel, en 2005 [INT08]. En 2006 Intel lanza sus microprocesadores Pentium D serie 900, que suponen el paso a la tecnología de 65 nm (los Pentium D de la serie anterior utilizaban la tecnología de 90 nm), y poco después presenta su nueva microarquitectura *Core*, orientada al desarrollo de microprocesadores multi-núcleo. En el último trimestre de 2008, está prevista la presentación de una nueva microarquitectura, la *Nehalem*, pensada para tecnologías de 45 nm a 32 nm, y con versiones para el mercado de los computadores de sobremesa, en 2008, y para los de servidores y portátiles, en 2009 y 2010 respectivamente. Estos microprocesadores incluirán entre cuatro y ocho núcleos de procesamiento con dos hebras por núcleo y frecuencias entre 2.66 y 3.2 GHz. Por otra parte, prácticamente todos los fabricantes de microprocesadores tienen *roadmaps* para el desarrollo de microprocesadores multi-núcleo. Así, en 2005 y 2007, Sun presentó los microprocesadores T1 y T2, respectivamente. Estos microprocesadores utilizan núcleos de procesamiento multihebra simultánea, incluyendo hasta 8 núcleos, y 32 y 64 hebras. En cuanto a

AMD, poco después de que apareciese el Pentium D de Intel, lanzó su primer microprocesador con dos núcleos, el Athlon 64 X2, para seguir en 2007 con su nueva microarquitectura K10 y microprocesadores de tres y cuatro núcleos como *Phenom*, de 2.4 a 2.6 GHz, y *Barcelona*, de 1.7 a 2 GHz, y tecnología de 65 nm.

La ley de Moore y los límites de crecimiento

No se puede hablar de la evolución en la densidad de integración, ni de la mejora de prestaciones de los procesadores sin hacer referencia a la ley de Moore. No se trata de una ley que se pueda deducir de las características de los procesos tecnológicos implicados en la fabricación de circuitos integrados. Todo parte de un artículo de Gordon E. Moore de 1965, en el número del trigésimo quinto aniversario de la revista *Electronics* [MOO65]. En este trabajo se indicaba que, desde el primer prototipo de microcircuito producido en 1959, cada año se venía doblando el número de componentes semiconductores de mínimo coste incluidos en un chip, y se extrapolaba este ritmo de integración observado hasta entonces. Lo importante de esta afirmación no es el ritmo de crecimiento que se indica, de hecho a lo largo del tiempo, ese ritmo ha pasado a ser del doble cada 18 meses al doble cada 24 meses, y existe cierta controversia acerca del mismo debido a la facilidad con que se pueden ajustar a distintos ritmos de crecimiento la nube de datos de integración (correspondientes a distintos fabricantes, etc.) que existen para un determinado periodo [TUO02]. Lo importante es que establece un ritmo de crecimiento exponencial para el aumento de la complejidad de los circuitos integrados. La ley de Moore pone de manifiesto cómo la influencia de un paradigma ampliamente aceptado contribuye a crear una imagen de lo verosímil, de lo posible y de lo real, y esta imagen influye de manera determinante en los proyectos de futuro y en la dirección de la innovación tecnológica.

El ritmo de aumento de la complejidad que se considera acertado en cada momento es el que utilizan las compañías para planificar el desarrollo de nuevos productos, la comunidad científica y

tecnológica para plantear los proyectos de investigación, e incluso las entidades públicas para planificar los programas de financiación de investigación y desarrollo. El tiempo de desarrollo de una nueva microarquitectura puede comprender varios años. Por ejemplo, el tiempo de diseño de la microarquitectura P6 de Intel (utilizada en el Pentum Pro, Pentium II y Pentium III) fue de unos cuatro años y medio. De hecho, según cuenta en su libro ([COL06]) Robert Colwell, ingeniero responsable de dicha microarquitectura, el proyecto de desarrollo de la P6 se inició en 1990, y el requisito era doblar las prestaciones de la microarquitectura P5 (utilizada en el Pentium), suponiendo el mismo proceso tecnológico de fabricación. Pero el Pentium, y por lo tanto la microarquitectura P5, ni siquiera estaba en el mercado en 1990, apareció en 1993.

Es importante tener una referencia de las prestaciones que deben ofrecerse en el momento de aparecer en el mercado, y esa referencia es la ley de Moore, que también se toma como índice de mejora de las prestaciones de los procesadores, aunque Moore se refiriese sólo al ritmo de crecimiento del número de transistores en el chip. Para una empresa, no alcanzar lo que marca la ley de Moore puede suponer una pérdida de cuota de mercado de fatales consecuencias. Por otra parte, plantearse un ritmo mayor que el establecido por la ley de Moore también es difícil de asumir, dado el considerable incremento en el volumen de inversión que eso supondría. Por tanto, la ley de Moore se ha convertido así en una especie de profecía autocumplida en la que los agentes que intervienen en el proceso se mueven bajo una especie de dilema del prisionero en el que nadie quiere correr el riesgo de perder el ritmo de los demás porque todo el mundo está convencido de las posibilidades de la tecnología de integración, que es la que establece la interconexión entre los distintos agentes (una interconexión en la que los agentes no intercambian productos sino ideas acerca de la tecnología y de sus oportunidades[LEN94]).

Pero en algún momento no se podrá seguir el ritmo que marca la ley de Moore. La pregunta no es si ese momento llegará sino cuándo: hasta cuándo se podrá mantener ese ritmo de crecimiento

exponencial por la influencia de factores, físicos o de otro tipo, imposibles de salvar. Para dar una respuesta existen diversas aproximaciones. Una de ellas es de tipo económico y proviene de la denominada segunda ley de Moore, según la cual el coste de desarrollo (incluyendo el coste de diseño) y fabricación de los circuitos integrados está creciendo a un ritmo del 25% anual, doblándose cada tres años. Así, una planta para integrar circuitos en obleas de 300 milímetros puede suponer una inversión de entre 2500 y 3500 millones de dólares, y requerir ventas de unos 6000 millones de dólares anuales. Sin embargo, hay que tener en cuenta que el año 2000 (el mejor año en cuanto a ventas de circuitos integrados) sólo 10 compañías estuvieron próximas a esos 6000 millones de dólares. Por otra parte, el aumento en el tamaño de los equipos de diseño de los microprocesadores ha crecido considerablemente. Por ejemplo, el equipo de diseño del 4040 de Intel estaba constituido por cuatro personas, el del Pentium en 1993 por 250, el de la microarquitectura P6 (Pentium Pro, Pentium II y Pentium III) en 1995 por 450, y el de la Netburst (Pentium 4) por 850 [SCH04]. A medida que los equipos de diseño crecen, los costes de coordinación y comunicación también son mayores. Todo esto ha generado una clara necesidad de cooperación entre las compañías, constituyendo alianzas para la fabricación compartida de sus circuitos, etc. Por ejemplo, AMD ha constituido alianzas con el Grupo UMC de Taiwan y con la alemana Infineon, y Sony, IBM y Toshiba se unieron para la fabricación del procesador Cell.

Otra predicción del final de la validez de la ley de Moore se puede obtener a partir del límite de Shanon/von Neumann/Landauer que se utiliza en el *modelo Gedanken* [ZHI03]. Aquí se tienen en cuenta las relaciones de incertidumbre de Heisenberg, y se sitúa en 1.5 nm el límite inferior para las dimensiones del transistor. Teniendo en cuenta el ritmo al que se están reduciendo éstas (aproximadamente un factor de 10 en 15 años), y que en 2008 ya se utiliza la tecnología de 45 nm en microprocesadores comerciales, el final para la estrategia de mejora de prestaciones basada en el paradigma de Moore se puede producir en menos de 20 años. De hecho, las predicciones del ITRS en su informe de 2007 indican que

en el año 2022 se integrarán transistores con una longitud física de puerta de 5 nm, sólo unas tres veces el límite inferior obtenido a partir del *modelo Gedanken*. Antes de ese momento habrá que tener clara la dirección a tomar. Si los recursos que ofrece la tecnología de integración no pueden mantener su ritmo de crecimiento, la pelota de la mejora de prestaciones de las máquinas queda en manos de los ingenieros de computadores y del desarrollo eficiente de software para estas nuevas plataformas. El valor añadido de una nueva arquitectura de computador eficiente crecería, y lo mismo sucedería con la capacidad de innovación en la ingeniería de computadores.

Las perspectivas futuras

Ante el agotamiento de la tecnología electrónica actual (basada en el estado de carga eléctrica de los dispositivos) para mantener el ritmo que marca la ley de Moore, se han planteado dos tendencias fundamentales. Por un lado está la búsqueda de nuevas tecnologías o dispositivos que sustituyan la actual. Dentro de esta tendencia se pueden ubicar los nuevos dispositivos que se siguen basando en el estado de carga para representar el estado computacional, como es el caso de los *nanotubos* [BIS05, HUT08]. No obstante, se estima que con ellos se podría conseguir una mejora de un factor de tres en tamaño o velocidad, sobre todo teniendo en cuenta que la densidad de dispositivos va a estar fundamentalmente limitada por la disipación energética (100 W/cm^2) y no por las dimensiones de los dispositivos. También se está investigando en tecnologías con nuevas variables de estado computacional como por ejemplo la conformación molecular, los *qubits* cuánticos, los dipolos magnéticos, etc. [HUT08]. En cualquier caso, la cuestión estriba en implementar dispositivos que utilicen esas variables de estado y conseguir que la nueva tecnología correspondiente sea capaz de escalarse en cuanto a densidad funcional, velocidad y consumo igual que lo ha hecho tecnología CMOS en los últimos 40 años.

La Tabla 1 proporciona algunas características de tecnologías alternativas en comparación con las de la tecnología CMOS. En ediciones anteriores a la última, el capítulo de dispositivos emergentes (*Emerging Research Devices*, *ERD*) del ITRS [ITRS] había concluido que ninguna de las tecnologías emergentes podría superar a la tecnología CMOS en el ámbito de la lógica booleana en sistemas de propósito general. En la última [ITRS07], se analizan las posibilidades de las tecnologías emergentes en el contexto de la tendencia actual y futura hacia circuitos integrados incluyendo núcleos de procesamiento heterogéneos. La ley de Amdahl justifica estas microarquitecturas heterogéneas [HIL08] ya que los distintos núcleos pueden utilizarse para acelerar de manera eficiente las diferentes partes de una aplicación, atendiendo a sus perfiles de cómputo característicos. Actualmente existen en el mercado ejemplos de microprocesadores multi-núcleo heterogéneos. Entre ellos están los *procesadores de red*, utilizados para acelerar el procesamiento de los protocolos de comunicación y de las aplicaciones de red [ORT08, CAS09], y el procesador Cell, inicialmente orientado al mercado de los juegos de ordenador (es el procesador de la PlayStation 3), pero también utilizado en plataformas para la computación de altas prestaciones [GUI08]. De hecho el *Roadrunner* de IBM, al que nos hemos referido, utiliza este microprocesador. La microarquitectura multi-núcleo *Havendale* prevista por Intel para 2009 incluirá una unidad de procesamiento de gráficos (GPU) [LIN08] junto a otros dos núcleos de procesamiento de propósito general.

En un futuro, estos microprocesadores multi-núcleo heterogéneos podrían incluir coprocesadores específicos, implementados mediante alguna de las tecnologías emergentes, para acelerar ciertas aplicaciones, entre las que se citan el reconocimiento del habla y de imágenes, o la minería de datos mediante motores de inferencia bayesianos [BOU08, CAV08]. Precisamente, esta alternativa para superar el límite de la ley de Moore, basada en el desarrollo de arquitecturas de propósito específico optimizadas para cada problema, implica de manera más directa a la ingeniería de computadores. Con el ritmo de mejora de prestaciones que marcaba

la ley de Moore, resultaba bastante arriesgado el esfuerzo de desarrollo de arquitecturas de propósito específico, cuyas prestaciones podían ser superadas por microprocesadores de propósito general de menor coste, y en un periodo de tiempo demasiado pequeño como para poder obtener beneficios con el sistema de propósito específico. Esta situación cambiaría si el ritmo de mejora de las prestaciones de los procesadores de propósito general fuera menor, o se estancase.

Tabla 1. Tecnologías alternativas a la CMOS [ITR07, BOU08]

Dispositivo	Transistor de efecto de campo (FET) CMOS	Estructuras 1D (nanotubos de carbono)	Transistor mono-electrónico (SET)	Dispositivos moleculares (transistor molecular, crossbar match)	Transistor de Spin
Densidad [dispositivos/cm^2]	2.8×10^{8} (10^{10})	4×10^{7} (4.5×10^{9})	2×10^{9} (6×10^{10})	2×10^{7} (2×10^{12})	10^{4} (4.5×10^{9})
Velocidad de Conmutación	1.5 THz (12 THz)	200 MHz (6.3 THz)	2 THz (10 THz)	100 Hz (1 THz)	--- (40 GHz)
Energía de conmutación [J]	10^{-16} (3×10^{-18})	10^{-11} (3×10^{-18})	$>1.3\times10^{-14}$ (10^{-17})	3×10^{-7} (5×10^{-17})	--- (3×10^{-18})
Rendimiento binario [Gbit/ns/cm^2]	1.6 (238)	10^{-8} (238)	2×10^{-4} (10)	2×10^{-9} (1000)	--- (---)

Valores alcanzados
(entre paréntesis valores máximos o mínimos considerados alcanzables para cada caso)

Por otra parte, los *roadmaps* de los fabricantes de microprocesadores para los próximos años muestran una clara tendencia a incluir más núcleos de procesamiento en los chips. Por ejemplo, Intel planea desarrollar una nueva microarquitectura denominada *Sandy Bridge*, con hasta 8 núcleos y 4 GHz, que estará disponible en 2010 para una tecnología de 32 nm, y en 2011 para 22 nm. Incluso se habla de una nueva microarquitectura denominada *Haswell*, disponible para 2012 con tecnología de 22 nm. En cuanto a AMD, a finales de 2008 y en 2009 aparecerán procesadores de 4 y 6 núcleos con tecnologías de 45 nm (microprocesadores *Shanghai* e *Instanbul*) y, para 2010, procesadores con 8 y 12 núcleos (microprocesadores *Sao-Paulo* y *Magny-Cours*), también con tecnología de 45 nm.

Dado que la mejora de prestaciones de los microprocesadores de propósito general se basa en la integración de más núcleos, el aprovechamiento de sus prestaciones necesita códigos paralelos eficientes. Surge así un punto de coincidencia con el desarrollo y la explotación eficiente de otras plataformas paralelas. Igual que en el caso de las arquitecturas de propósito específico, hasta ahora, el ritmo de crecimiento de las prestaciones de los procesadores ha establecido importantes restricciones en el tiempo máximo disponible para el desarrollo y la explotación de computadores paralelos. Pero ¿qué pasará cuando este ritmo de mejora de la velocidad de los procesadores disminuya?. Entonces habrá que familiarizarse con el procesamiento paralelo pero, como ya se ha dicho, está demanda ya está presente hoy, con los procesadores multi-núcleo.

La escalabilidad es crucial para las plataformas paralelas. Sobre todo en el caso de que se quiera aprovechar el trabajo paralelo del número tan elevado de procesadores que se utilizan en los supercomputadores actuales. Por ejemplo, el computador *Roadrunner* de IBM dispone de unos 122.000 procesadores. En este tipo de plataformas paralelas, la comunicación constituye un límite esencial, sobre todo teniendo en cuenta que presentan dimensiones del orden de las decenas de metros. Por ejemplo, el *Roadrunner* ocupa una superficie de 5.200 pies^2. Eso significa distancias entre nodos que pueden llegar hasta los treinta metros. El tiempo mínimo para comunicar dos nodos separados por treinta metros sería de unos 0.1 microsegundos (teniendo en cuenta la velocidad de la luz), es decir, unos 300 ciclos con procesadores funcionando a 3 GHz. Para que sea posible obtener una eficiencia del 75% en una plataforma con p procesadores harían falta más de $900p$ ciclos de computación por cada comunicación entre procesadores distantes. En una plataforma con 100.000 procesadores, eso supondría más de 90 millones de ciclos de cómputo distribuidos entre los procesadores antes de que sea necesario que se comuniquen. Dado que los procesadores pueden terminar varias instrucciones por ciclo, debería haber tiempo para procesar casi 200 millones de instrucciones entre comunicaciones, aún con esta estimación tan extraordinariamente

optimista en la que se ha utilizado la velocidad de la luz para estimar los tiempos mínimos.

Así pues, es difícil alcanzar eficiencias elevadas en problemas con necesidades de cómputo elevadas que pretendan aprovechar computadores con un número elevado de procesadores. Como ejemplo se puede considerar las simulaciones de la dinámica de moléculas biológicas, a la que nos referimos anteriormente. Una biomolécula se describe a nivel de átomos a través de la función de potencial molecular que incluye los términos de energía correspondientes a la desviación en las longitudes, los ángulos y la torsión de los enlaces, junto con las interacciones electrostáticas y de van der Waals (término correspondiente a fuerzas no acotadas):

$$E = E_{enlaces} + E_{angulos} + E_{dihedros} + E_{no\_acotada}$$

Los distintos entornos de simulación de dinámica molecular se diferencian en detalles de las expresiones de la función de energía (diferentes valores para los parámetros). La minimización de la función de energía, *E*, proporciona una imagen estática útil para comparar distintas configuraciones, pero si se quiere información de los procesos dinámicos, como ocurre en el caso del problema del plegamiento de proteínas, hay que resolver la ecuación del movimiento de Newton ($F=m.a$), donde las fuerzas sobre los átomos se obtienen a partir del gradiente negativo de la función de energía potencial. En los sistemas biológicos las interacciones no-acotadas (van der Waals y electrostáticas) son las más costosas y las que requieren mayor volumen de comunicación. Por otra parte, es necesario incluir el disolvente (usualmente agua) en la simulación. Existe un procedimiento *explícito*, en el que se consideran las moléculas del disolvente con algún tipo de simplificación, o *implícito*, cuando el disolvente se trata como un medio continuo y se promedian sus propiedades. Entre las técnicas explícitas está la técnica PME (*Particle Mesh Ewald*), y entre las implícitas la GB (*Generalizad Born*). Estas técnicas suelen requerir comunicaciones colectivas que involucran procesadores no conectados directamente. En [ALA06] se pone de manifiesto la dificultad que implica

aprovechar el paralelismo en las simulaciones de dinámica molecular. Por ejemplo, el procedimiento *HhaI*, que simula un complejo proteína-DNA de 61.641 átomos mediante método explícito PME para el disolvente, alcanza una ganancia de velocidad de sólo 14.02 con 128 procesadores (una eficiencia inferior al 11%). En el caso de *RuBisCO*, un modelo de la enzima Ribulosa Bifosfato Carboxylasa/Oxigenasa con 73.920 átomos y el método implícito GB para el disolvente, se alcanza una ganancia de velocidad de 279.31 con 2048 procesadores (una eficiencia inferior al 14%).

Por tanto, se deben buscar modelos de cómputo y algoritmos que se ajusten a los requisitos de cómputo y comunicación y que presenten la suficiente localidad en el acceso a los datos para evitar comunicaciones entre procesadores distantes. Un ejemplo de esta búsqueda de métodos alternativos adaptables a las plataformas masivamente paralelas lo constituye la utilización de autómatas celulares para resolver sistemas de ecuaciones diferenciales en derivadas parciales [DON01]. Se trata de determinar el autómata celular cuya dinámica se ajusta a la de los elementos del sistema físico para que el comportamiento macroscópico que se desea conocer emerja de la interacción de los elementos del autómata celular, que interactúan utilizando comunicación local y reglas usualmente sencillas [TOF99]. Otra alternativa es la búsqueda de procedimientos que requieran únicamente una comunicación asíncrona entre tareas paralelas para relajar las necesidades de sincronización entre procesadores distantes. En esta línea se encuentran los algoritmos evolutivos paralelos [ALB02]. Gracias a ellos, la resolución de muchos problemas de optimización complejos se puede abordar con procedimientos dotados de un paralelismo implícito que evita patrones irregulares de comunicación [CAL09].

Las plataformas paralelas plantean algunos interrogantes más. Entre ellos está su coste, no sólo el asociado a su fabricación e instalación sino también el coste de mantenimiento. El consumo energético de 120.000 procesadores no es en absoluto despreciable. La necesidad de refrigeración para evacuar el calor generado establece requisitos importantes en los centros de supercomputación,

además del consumo energético que supone. Igual que existe un TOP500 también existe un Green500 [GRE08] en el que se tiene en cuenta el consumo por unidad de prestaciones. Un computador como el IBM Blue Gene/L [GAR05], el número 1 en el TOP500 hasta la última edición de 2007, consume 2500 KW y proporciona 112 MFLOPS/W. En la edición de Junio de 2008 la situación ha cambiado. La mejor marca en el Green500 corresponde a 488.14 MFLOPS/W, y el primer computador en el TOP500 está en la tercera posición del Green500, proporcionando 437.43 MFLOPS/W, y consumiendo unos 2400 KW. Es decir, un consumo parecido al del Blue Gene/L, con una eficiencia energética casi cuatro veces mayor.

Conclusión

Parece que los proyectos de ingeniería de computadores presuponiendo mejoras sin límite en la capacidad de integración no van a ser factibles mucho más allá del 2025, a no ser que entre tanto se produzca un *patrón de innovación de transformación*, similar al que tuvo lugar con la invención del transistor en los años 40.

Mientras eso no ocurra, la innovación en los computadores vendrá de la mano de las arquitecturas de propósito específico en microprocesadores multi-núcleo heterogéneos y del aprovechamiento eficiente del paralelismo en sus distintos niveles, tanto en el microprocesador como en los servidores. Igual que la escasez y los precios elevados de combustible hacen más relevante el desarrollo de motores eficientes en cuanto a su consumo, los límites en el número de transistores disponibles harán decisiva la optimización de los recursos de cómputo en las aplicaciones. Para conseguir códigos eficientes en plataformas que pueden ser muy diversas se necesita realizar un trabajo considerable de desarrollo de herramientas para la programación paralela (lo ideal sería disponer de compiladores paralelos) y de algoritmos paralelos con relaciones elevadas de computación/comunicación, o patrones locales de comunicación.

Por otra parte, el desarrollo de plataformas de cómputo y de programas eficientes en cuanto al aprovechamiento de energía, constituirá un desafío importante, tanto desde el punto de vista del hardware como del software.

Bibliografía

[AGA00] Agarwal, V.; et al.;"Clock Rate versus IPC: The End of the Road for Conventional Microarchitectures". Proc. 27th Ann. Int'l Symp. Computer Architecture, ACM Press, pp.248-259, 2000.

[ALA06] Alam, S.R.; Vetter, J.S.; Agarwal, P.K.; Geist, A.:"Performance Characterization of Molecular Dynamics Techniques for Biomolecular Simulations". ACM PPoPP'06, pp.59-68, 2006.

[ALB02] Alba, E.: "Parallel evolutionary algorithms can achieve super-linear performance". Information Processing Letters, 82, pp.7-13, 2002.

[AMD67] Amdahl, G.:"Validity of the single processor approach to achieving Large-Scale Computing Capabilities". AFIPS Conference Proceedings, 30, pp. 483-485, 1967.

[BIS05] Bishop, D.:"Nanotechnology and the end of Moore's Law?". Bell Labs Technical Journal, 10 (3), pp.23-28, 2005.

[BOU08] Bourianoff, G.; et al.: "Boolean logic and alternative information-processing devices". IEEE Computer, pp. 38-46. Mayo, 2008.

[CAL09] Calvo, J.C.; Ortega, J.:"Parallel protein structure prediction by multiobjective optimization". 17th Euromicro Int. Conference on Parallel, Distributed and Network-based processing, PDP 2009.

[CAS09] Cascón, P.; Ortega, J.; Haider, W.; Díaz, A. F.; Rojas, I.:"A Multi-threaded network interface using network processors". 17th Euromicro Conference on Parallel, Distributed and Network-based Processing, PDP 2009

[CAV08] Cavin, R.; et al.: "Emerging Research Architectures". IEEE Computer, pp. 33-37. Mayo, 2008.

[CLA88] Clark, N.:"Some new approaches to evolutionary economics". Journal of Economic Issues, Vol.XXII, No.2, 1988.

[COL06] Colwell, R.P.:"The Pentium Chronicles". IEEE Computer Society, Wiley Inter-Science, 2006.

[DON01] Dongarra, J.; Walter, D.W.:"The Questing for Petascale Computing". Computing in Science & Engineering, pp. 32-39. Mayo/Junio, 2001.

[ELZ94] Elzen, B.; Mackenzie, D.:"The Social Limits of Speed: The development and use of Supercomputers". IEEE Annals of the History of Computing, Vol.16, No.1, pp.46-61, 1994.

[FLY98] Flynn, M.J.:"Computer Engineering 30 years after the IBM Model 91". IEEE Computer, pp.27-31, Abril, 1998.

[GAL98] Galvin, R.:"Science and Technology Roadmaps". Science, vol.280. Mayo, 1998.

[GAR05] Gara, A., et al.:"Overview of the Blue Gene/L System Architecture". IBM J. Rresearch and Development, Vol. 49, No.2/3. Marzo/Mayo, 2005.

[GOR08] Gorton, I.; et al. :"Data-intensive computing in the 21st Century". IEEE Computer, pp.30-32. Abril, 2008.

[GRE08] Green500 (The Green500 list): http://www.green500.org/lists. Junio, 2008.

[GUI08] Guizo, E.:"Solving the Oil equation". IEEE Spectrum, Vol. 4, No.1, pp.24-28. Enero, 2008.

[HEN03] Hennessy, J.L.; Patterson, D.A.:"Computer Architecture. A Quantitative Approach" (3ª Edición). Morgan Kaufmann Pub., 2003.

[HIL08] Hill, M.D.; Marty, M.R.:"Amdahl's Law in the Multicore Era". IEEE Computer, pp.33-38. Julio, 2008.

[HUT08] Hutchby, J.A.; et al.: ”Emerging Nanoscale Momory and Logic Devices: A Critical Assessment”. IEEE Computer, pp.28-32. Mayo, 2008.

[INT08] Intel processors quick referente guide: http://www.intel.com/pressroom/kits/quickreffam.htm

[ISA00] Isaac, R.: Entrevista de R.R. Schaller (recogida en [SCH04]), 2000.

[ITRS] Internacional Technology Roadmap for Semiconductors: http://www.itrs.net/

[ITRS07] International Technology Roadmap for Semiconductors, 2007 Executive summary: http://www.itrs.net/Links/2007ITRS/ExecSum2007.pdf.

[KLI86] Kline, S.J.; Rosenberg, N.:”An overview of innovation”. En “The positive sum strategy: harnessing technology for economic growth”, R. Landau y N. Rosenberg (Eds.), Academic Press, 1986.

[LEN94] Lente, H. van; Rip, A.: ”Expectations in technological developments: an example of prospective structures to be filled by agency”. 13th Worl Congress Sociology, ISA, 1994.

[LIN08] Lindholm, E.; Nickolls, J.; Oberman, S.; Montrym, J.:”NVIDIA Tesla: A unified graphics and computing architecture”. IEEE Micro, 28 (2), pp. 39-55 . Marzo/Abril 2008.

[MOO65] Moore, G.E.:”Cramming more components onto integrated circuits”. Electronics, Vol.38, No.8, 1965.

[MUD01] Mudge, T.:”Power: A First-Class Architectural Design Constraint”. IEEE Computer, Vol. 34, No.4, pp.52-58. Abril, 2001.

[NGO92] Ngo, T.; Marks, J.:”Computational Complexity of a Problem in Molecular-Structure”. Protein Engineering, 5(4), pp.313-321, 1992.

[ORT05] Ortega, J; Anguita, M.; Prieto, A.: “Arquitectura de Computadores”. Ed. Thomson, 2005.

[ORT08] Ortiz, A.; Ortega, J.; Díaz, A.F.; Cascón, P.; Prieto, A.:"Protocol offload analysis by simulation". Journal of System Architecture, 2008.

[RYC99] Rycroft, R.W.; Kash, D.E.:"The complexity challenge: technological innovation for the 21st century", Pinter, Londres, 1999.

[SCH04] Schaller, R.R.:"Technological innovation in the Semiconductor Industry: A case study of the International Technology Roadmap for Semiconductors (ITRS)". Tesis Doctoral, George Mason University, 2004.

[TOF99] Toffoli, T.: "Programmable Matter Methods". Future Generation Computer Systems, Vol. 16, No. 2-3, pp. 187-201. Diciembre, 1999.

[TOP08] TOP500 (The TOP500 list): http://www.top500.org/. Junio 2008.

[TRE95] Tredennick, N.:"Technology and Business: forces driving microprocessor evolution". Proceedings of the IEEE, Vol.83, No.12, pp.1641-1652. Diciembre, 1995.

[TUO02] Ilkka Tuomi: "The Lives and Death of Moore's Law". URL: http://firstmonday.org/issues/issue7_11/tuomi/index.html. Noviembre, 2002

[VAJ01] Vajapeyam, S.; Valero, M.: "Early 21st Century Processors". IEEE Computer, Vol.34, No. 4, pp. 47-50. Abril 2001.

[XUE08] Xue, Y.; et al.: "Quantitative retrieval of Geophysical Parameters using Satellite Data". IEEE Computer, pp. 33-40. Abril, 2008.

[ZHI03] Zhirnov, V.V.; et al.: "Limits to Binary Logic Switch Scaling – A Gedanken Model". Proceedings of the IEEE, pp.1934-1939. Noviembre, 2003.