Anton 3: Twenty Microseconds of Molecular Dynamics Simulation Before Lunch

David E. Shaw,[a] Peter J. Adams,[b] Asaph Azaria, Joseph A. Bank, Brannon Batson, Alistair Bell, Michael Bergdorf, Jhanvi Bhatt,[b] J. Adam Butts, Timothy Correia, Robert M. Dirks,[c] Ron O. Dror,[b] Michael P. Eastwood, Bruce Edwards, Amos Even, Peter Feldmann, Michael Fenn, Christopher H. Fenton, Anthony Forte, Joseph Gagliardo, Gennette Gill, Maria Gorlatova,[b] Brian Greskamp, J.P. Grossman,[b] Justin Gullingsrud, Anissa Harper, William Hasenplaugh,[b] Mark Heily, Benjamin Colin Heshmat, Jeremy Hunt, Douglas J. Ierardi, Lev Iserovich, Bryan L. Jackson, Nick P. Johnson, Mollie M. Kirk,[b] John L. Klepeis, Jeffrey S. Kuskin, Kenneth M. Mackenzie, Roy J. Mader, Richard McGowen,[b] Adam McLaughlin, Mark A. Moraes, Mohamed H. Nasr,[b] Lawrence J. Nociolo, Lief O'Donnell, Andrew Parker, Jon L. Peticolas, Goran Pocina, Cristian Predescu, Terry Quan, John K. Salmon,[c] Carl Schwink,[b] Keun Sup Shim, Naseer Siddique,[b] Jochen Spengler,[b] Tamas Szalay, Raymond Tabladillo,[b] Reinhard Tartler,[b] Andrew G. Taube, Michael Theobald, Brian Towles,[b] William Vick,[b] Stanley C. Wang, Michael Wazlowski, Madeleine J. Weingarten, John M. Williams,[b] and Kevin A. Yuh

D. E. Shaw Research, New York, NY 10036, USA

***Abstract*—Anton 3 is the newest member in a family of supercomputers specially designed for atomic-level simulation of molecules relevant to biology (e.g., DNA, proteins, and drug molecules). Anton 3 achieves order-of-magnitude improvements in time-to-solution over its predecessor, Anton 2 (the current state of the art), and is over 100-fold faster than any other currently available supercomputer, thereby enabling broad new avenues of research on critical questions in biology and drug discovery. This speedup means that a 512-node Anton 3 simulates a million atoms at over 100 microseconds per day. Furthermore, Anton 3 attains this performance while consuming an order of magnitude less energy per simulated microsecond than any other machine. Like its predecessors, Anton 3 was designed from the ground up around a new custom chip to best exploit the capabilities offered by new technologies. We present here the main architectural and algorithmic developments that were necessary to achieve such significant advances.**

I. Justification for ACM Gordon Bell Prize

Anton 3 sets new records (by orders of magnitude) in the speed of all-atom biomolecular simulations. Compared with the fastest general-purpose supercomputers, a 64-node Anton 3 is 120 times faster on a 24,000-atom standard benchmark, while a 512-node machine is 460 times faster on a 2.2-million-atom ribosome system.

II. Performance Attributes

Category of achievement	Time-to-solution
Type of method used	Explicit
Results reported on basis of	Whole application including I/O
Precision reported	Mixed precision
System scale	512 nodes
Measurement mechanism	Timers

III. Overview of the Problem

All-atom molecular dynamics (MD) simulation is a powerful tool for studying the behavior of biological macromolecules (e.g., proteins, RNA, and DNA) [1][2]. It can answer fundamental questions about the function of these biomolecules by simulating their three-dimensional motion, and can facilitate drug design by identifying and quantifying how small-molecule compounds interact with potential drug targets. Performing MD simulations at the scale and speed necessary to address these research questions within practical timelines, however, remains a highly challenging problem in supercomputing.

We previously designed, built, and deployed two generations of Anton supercomputers, Anton 1 [3][4] and Anton 2 [5], to tackle this problem, and when each debuted, it reached one hundred times greater speeds on MD simulations than the fastest general-purpose supercomputers of the time. These increases in performance have translated into significant scientific advances, as reported in over 300 peer-reviewed papers from hundreds of research groups granted time on these machines [6]. Furthermore, simulations from these machines have contributed to the discovery of three drug candidates currently in clinical trials.

Continued advances in the performance of MD simulations could be extremely valuable, as some important biological systems (including many that are highly relevant to drug design) comprise millions of atoms, while many scientifically and pharmaceutically important structural changes occur on timescales that have not historically been accessible to simulation. Moreover, accurately obtaining statistical properties of biochemical systems (e.g., binding rates of drugs to their targets) can require milliseconds of aggregate simulated time.

The ability of previous Anton machines to access long timescales on complex systems has been instrumental in our drug discovery efforts. An example of such a simulation is illustrated in Fig. 1, which shows a snapshot from a 185,000-

[a] David E. Shaw is also affiliated with the Department of Biochemistry and Molecular Biophysics, Columbia University, New York, NY 10032.
E-mail correspondence: David.Shaw@DEShawResearch.com.
[b,c] See Acknowledgements section for further information about these former D. E. Shaw Research team members.

SC '21, November 14–19, 2021, St. Louis, MO, USA

ACM ISBN 978-1-4503-8442-1/21/11.
https://doi.org/10.1145/3458817.3487397

atom, 24-microsecond Anton 2 simulation of a potassium ion channel (a family of proteins we have simulated extensively as part of our drug discovery research). We see many new research opportunities, however, that require faster MD simulation, and many interesting biological systems that fall outside the capacity of Anton 2. Consequently, we developed Anton 3, a special-purpose supercomputer designed around a new custom chip to once again dramatically increase the speed and scale of all-atom MD simulations relevant to basic science and drug design.

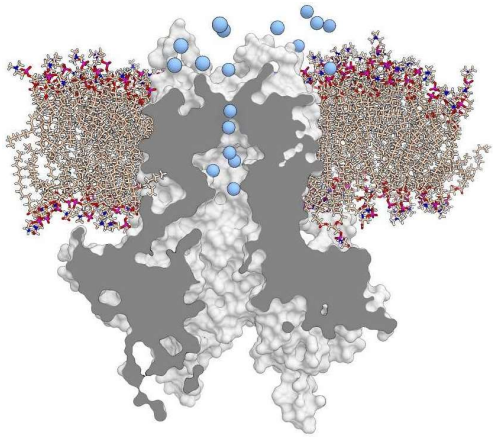


Fig. 1. Rendering from an MD simulation of potassium ions (blue spheres) passing through an ion channel protein (gray cutaway) embedded in a cell membrane (the fully rendered molecules on either side of the channel).

MD simulation predicts the three-dimensional movements of atoms in a chemical system over a large number of discrete time steps. During each time step, inter-atomic forces among the atoms are computed using physics-based models known as force fields. These inter-atomic forces consist of *bond terms* that model forces between small groups of atoms usually separated by 1–3 covalent bonds, and *non-bonded* forces between all remaining pairs of atoms. The forces on a given atom are summed to give a total force on the atom, which (by Newton's second law) directly determines the acceleration of the atom and thus (by integrating over time) can be used to update the atomic positions and velocities to their values at the next time step. Without approximating some calculations, the number of inter-atomic forces computed on each time step scales quadratically with the number of atoms, meaning there is an enormous increase in time-to-solution with increasing system size. Furthermore, time steps on the order of a femtosecond are required for stable and accurate integration; around a billion time steps are thus necessary to simulate a microsecond of atomic motions.

In order to make such simulations computationally tractable, most implementations of MD (including those on Anton machines) express the forces among non-bonded atoms as a sum of range-limited forces and long-range forces. Range-limited forces decay rapidly with distance, and are individually computed between pairs of atoms up to a cutoff distance. Long-range forces, which decay more slowly with distance, are computed using a range-limited pairwise interaction of the atoms with a regular lattice of grid points, followed by an on-grid convolution, followed by a second range-limited pairwise interaction of the atoms with the grid points [7][8][9]. With this optimization, the number of operations in the force calculation grows nearly linearly with the number of atoms (instead of quadratically).

Each machine in the Anton family executes the MD algorithm entirely within custom chips we designed for that machine. Fig. 2 shows the dataflow of an MD simulation, indicating the division of labor among special-purpose pipelines and general-purpose processors within the new Anton 3 chip. The figure is a simplified representation of the computation; the functionality shown in each element of the figure is often distributed across several functional units, some of differing types. The force summation, for example, is implemented as a distributed hardware reduction, and the non-bonded forces are computed in two different types of specialized pipelines.

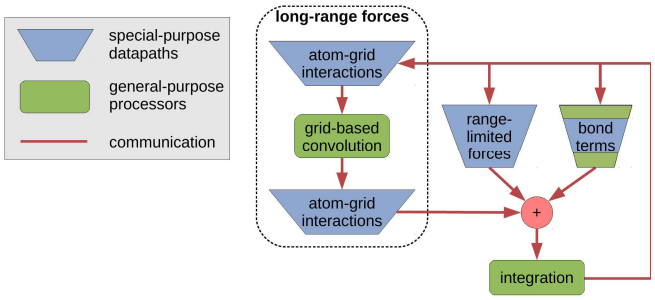


Fig. 2. Dataflow of an MD simulation. On Anton 3, range-limited non-bonded forces, simpler bond terms, and atom-grid interactions are computed by special-purpose datapaths. Grid-based convolution, more complex bond terms, and integration are computed by general-purpose processors.

Approximations are also used to reduce computational demand. With appropriate integration algorithms [10], for example, the costly long-range forces can be computed on only every second or third time step. In addition, the use of rigid constraints [11][12] to eliminate the fastest motions of hydrogen atoms allows time steps up to ~2.5 femtoseconds. If the masses of hydrogen atoms are also artificially increased, time steps as long as 4–5 fs [13][14] are possible. About 10^4 numerical operations are required per atom for each time step, which translates to around 10^{18} operations per microsecond of simulated time on a system of one million atoms, even combining these optimizations and approximations. This computational intensity, together with the fine-grained communication needed within each time step to parallelize the computation [15], makes MD simulation of biomolecular systems an extreme challenge in high-performance computing.

IV. Current State of the Art

The utility of MD simulation of biomolecular systems ultimately depends on simulation speed, which is reported here in microseconds of simulated time per day of elapsed wall-clock time. To simplify comparisons in this section, we focus on simulations of a million-atom system, the satellite tobacco mosaic virus (STMV), both because it is representative of many biologically interesting systems and because it has been used extensively to benchmark performance on many machines and software packages. We report comparative performance on a wider range of benchmarks in Section VII.

MD simulations are frequently run on commodity computers, with one or more GPUs serving as computing accelerators [16]. Such single-node platforms are supported by many MD software packages (e.g., ACEMD [17], AMBER [18], CHARMM [19], Desmond [20], GROMACS [21], HOOMD-blue [22], LAMMPS [23], NAMD [24], and OpenMM [25][26]), each with different features and performance characteristics.

At present, the fastest single-node commodity platform for these codes contains eight NVIDIA A100 SXM4 GPUs; the peak performance reported for this platform on the STMV benchmark is under 0.07 µs/day [20][21][25]. Attempting to reduce time-to-solution on such platforms by increasing the number of processing units above eight yields no improvement on the STMV benchmark [21]. Scaling is even worse for smaller molecular systems because the cost of distributing computations among the large number of GPU cores and integrating the results of those computations on each time step overshadows any gains from parallelization across the cores [27].

Several MD software packages (GENESIS [28], GROMACS, HOOMD-blue [29], LAMMPS, and NAMD) can scale simulation of biomolecular systems with tens of millions to a few billion atoms across machines comprising large numbers of general-purpose nodes, but such scaling on these machines has not resulted in performance beyond 0.2 µs/day [30]. The resulting simulations offer short-timescale (sub-microsecond) views of extremely large biomolecular systems, but they do not achieve the performance necessary for long-timescale (multi-microsecond to millisecond) simulation of systems with ten million atoms or fewer, a size range that includes myriad biomolecules of significant interest for basic scientific research and drug discovery.

Aside from the Anton machines, other supercomputers specialized for MD simulation have recently been reported. MDGRAPE-4A [31] is a 512-node special-purpose machine built with one custom chip per node. The capacity of MDGRAPE-4A has not yet been reported, but its estimated performance on STMV (extrapolating from a simulation of ~100,000 atoms) [32] would be ~0.1 µs/day. FPGA-based specialized hardware [33][34] has also been designed for MD, with reported performance competitive with that of single GPUs.

Anton 2 is the current state of the art for all-atom MD simulation of biochemical systems of up to three million atoms. It is typically used to perform simulations for tens to hundreds of microseconds each (though some have reached a few milliseconds) and is the foundation of our drug discovery and biochemistry research. Our 512-node Anton 2 machines (with a single custom-designed chip per node) achieve around 10 µs/day of simulation on the million-atom STMV benchmark, which at the time of their deployment in 2013 represented a speedup of over two orders of magnitude over what were then the fastest general-purpose supercomputers available, and remain over an order of magnitude faster than today's top supercomputers. The Anton 3 supercomputer described in this paper redefines the state of the art for time-to-solution as well as per-node throughput on MD simulation, thanks to significant innovations in its new design.

V. Innovations Realized

This section first presents an overview of the Anton 3 chip architecture, and then details selected innovations in subsections A–D.

Anton 3 was designed from the ground up to reduce time-to-solution by maximizing parallelism and exploiting evolving hardware technology. Though it carries over a few general design themes from Anton 1 and Anton 2—including the use of specialized chips with an integrated 3D torus network (Fig. 3a), and the co-design of algorithms, hardware, and software—the actual design of Anton 3 takes very few details from its predecessors. The novel design of the Anton 3 chip supports many new molecular simulation capabilities and is optimized to address two key scaling trends in semiconductor technology: increased power consumption of high-performance chips and reduced communication bandwidth relative to computational throughput, both of which make it difficult to increase the effective parallelization of the MD application.

As in prior generations, the MD application partitions the 3D space of the chemical system into contiguous *boxes* and assigns them to nodes so that neighboring chemical system boxes reside on neighboring nodes in the torus. Each node calculates forces between atoms within its assigned box (*Home Box*), and also sends the positions of a subset of these atoms to other nodes to allow computation of the forces between atoms in different boxes. The resulting forces are then returned to their originating node, which subsequently integrates the forces to update atom positions for the next time step.

Although effective parallelization of MD depends on maintaining a balance between communication and computation, communication bandwidth is scaling more slowly than computational throughput in current semiconductor technologies. To address this challenge, the Anton 3 chip adopts a highly regular, tiled layout (Fig. 3b) that optimizes both communication and computation. The chip consists primarily of repeated *Core Tiles* and *Edge Tiles*. Core Tiles are arranged in an array of 12 rows by 24 columns in the center of the chip, and contain both specialized pipelines and general-purpose processors to perform the MD computations. Edge Tiles flank the Core Tile array on the left and right, managing communication between Core Tiles and the inter-chip 3D torus network (using off-chip high-speed serial network links outboard of the Edge Tiles in Fig. 3b). The chip contains 96 off-chip serial lanes (SERDES transmit/receive pairs) operating at 29 Gbps in each direction, providing 5.6 Tbps of total bandwidth.

Fig. 3c shows the components of the Core Tile in more detail. A network router (*Core Router*) connects the computational blocks in the tile to a general-purpose 2D mesh network-on-chip. Dedicated *Position Buses* and *Force Buses* also traverse the tile carrying atom positions and forces to and from the *Pairwise Point Interaction Modules* (PPIMs), which contain the specialized pipelines for computing non-bonded interactions (subsection B). New in Anton 3, a specialized *Bond Calculator* (BC) handles computation of bond term forces (subsection D). The two *Geometry Cores* (GCs), and their associated 128-KiB *Flex SRAM* memory blocks, handle all time-step processing that is not performed by the BC or PPIM,

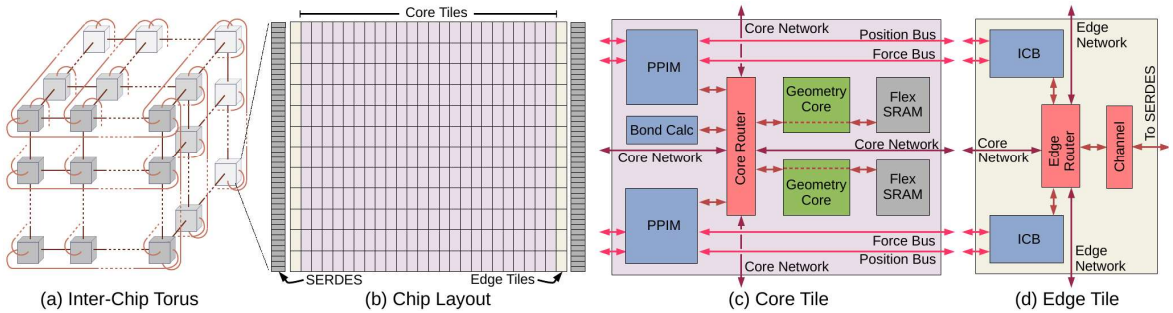


Fig. 3. (a) Inter-chip 3D torus network, (b) tiled layout of the Anton 3 chip showing Core Tiles (purple) and Edge Tiles (yellow), (c, d) block diagrams of the Core Tile and Edge Tile.

including sequencing of data movement, grid convolutions for distant interactions, integration of atom positions, and computation of the few interactions that are too complex for the specialized pipelines. The microarchitecture of the GC is optimized for 3D geometry and MD computation, while maintaining high performance per unit area on general-purpose code. To ensure high manufacturing yield for the large die, Anton 3 can circumvent manufacturing defects in Core Tiles by operating with an arbitrary number of Core Tile columns disabled; results in this paper reflect 22 enabled Core Tile columns.

The Edge Tiles in Fig. 3d contain the logic for the off-chip links (*Channels*), with each Channel connecting to one of the chip's six neighbors in the 3D torus using a group of SERDES. Each Channel also connects to an *Edge Router*, which forms an *Edge Network* with the other Edge Tiles on the same edge of the die, allowing traffic to turn across dimensions in the inter-chip 3D torus network. The Edge Router also connects to the Core Tile's 2D mesh network for injection and ejection. Finally, the *Interaction Control Blocks* (ICBs) connect the Edge Router to the Force and Position Buses that run across the Core Tile array. The ICBs include large buffers and programmable DMA engines, which are used to send atom positions onto the Position Buses. They also receive atom forces from the Force Buses and send them to the Edge Network for delivery to the Flex SRAMs.

A. New Custom Network

On-chip networks are heavily constrained by the semiconductor technology, particularly by the poor scaling of wire delay and density with respect to logic. In previous technology generations, fast, low-latency wiring resources were more readily available, and it was possible to make reasonable connections between on-chip subsystems in an unstructured way by auto-routing them through or around physical blocks. In current technologies, this is no longer practical when the highest performance is required. Anton 3's tiled layout benefits on-chip communication by allowing network links to exploit repeated, highly optimized routes between tiles, resulting in high bandwidths and low latencies for the on-chip networks.

Physical packaging constraints also shaped the design of off-chip networks, and have held the number of SERDES lanes per node roughly constant across Anton generations. Constraints on long-range electrical signaling also limit per-lane bandwidth. To achieve the required bandwidth scaling, Anton 3 introduces a low-latency MD-specific compression algorithm at the core side of the SERDES. Both the compressor at the sending side and the decompressor at the receiving side cache the most recent atom positions sent over the link, and use the past positions to implement a second-order predictor (incorporating position, velocity, and acceleration) for each atom's current position. For atoms in the cache, the compressor then sends only the difference between the predicted and actual positions, which is of small magnitude and can be compactly encoded using a variable-length format in fewer bits than the full position. The same variable-length format is also used for forces.

These compression techniques increase the effective bandwidth available to the application by a factor of two. Together with a doubling of Anton 2's SERDES bitrate (Table 1), the Anton 3 off-chip network provides a fourfold increase in bandwidth available to the application over Anton 2. In addition, a new parallel algorithm (subsection C) reduces the bandwidth demand of the application. This combination of innovations prevents torus bandwidth from limiting Anton 3 performance.

B. Specialized Pairwise Interactions at Different Precisions

Most of the computational work in MD involves range-limited pairwise interactions between atoms. Each atom must interact with all other atoms within the cutoff radius R_{cut}, and each interaction involves computing various types of forces. As in previous Anton generations, Anton 3 computes these forces in PPIMs, each comprising several *Pairwise Point Interaction Pipelines* (PPIPs). But unlike previous generations, Anton 3 further specializes the PPIPs into two types—*Big PPIPs* and *Small PPIPs*—to exploit the relationship between force and the interaction distance (r). Forces decay with distance, resulting in large forces for small r (interactions between nearby atoms) and smaller forces for larger r (interactions between atoms that are farther apart).

The volume of a spherical shell of radius r and thickness t is $4\pi\,(r^3-(r-t)^3)/3$, and thus there are substantially fewer nearby interactions ($r \to 0$) than there are farther-away interactions ($r \to R_{cut}$), as shown in Fig. 4a. Anton 3 exploits this fact by mapping nearby interactions to a Big PPIP with 23-bit datapaths and farther interactions to a Small PPIP with reduced-width 14-bit datapaths. Even though the datapath widths differ greatly between the two pipelines, the numerical errors in their force calculations are roughly equal in magnitude, as only larger forces require the wider datapath to achieve a given numerical precision.

An R_{cut} of 8 Å, for example, provides acceptable accuracy for many common types of force calculations. With that setting, the Small PPIP typically handles interactions with $r > 5$ Å and the Big PPIP handles those with $r \leq 5$ Å. The resulting ratio of Small to Big interactions is $(8^3 - 5^3)/\,5^3 \approx 3$. For any given simulation, the Anton 3 PPIMs compute a variety of functions with a range of R_{cut} values, but modeling suggests and experience has proven the 3:1 ratio of Small to Big PPIPs to be near-optimal.

The Small PPIPs are further specialized in that they can compute only the most common types of interactions. Consequently, control and data storage elements of their pipelines are simplified. Moreover, datapath area scales superlinearly with bit width, w, because narrower units contain fewer logic gates ($O(w^2)$ for multipliers and $O(w \log w)$ for adders). The simpler logic in the Small PPIP also requires fewer pipeline registers. The combination of these area savings allows three Small PPIPs to fit into the same area as one Big PPIP. Anton 3 thus fits four PPIPs' worth of interaction throughput (one Big and three Small) in the same area as would have been required for two Big PPIPs.

Splitting the Big and Small PPIPs offers another benefit in supporting new types of force calculations. Each extra force calculation option incurs silicon area overhead (e.g., new arithmetic units, muxes, and wires), but most options of interest are either invoked infrequently or only relevant for short-range interactions. This means they can be implemented solely in the Big PPIP, halving their effective cost. Perhaps counterintuitively, increasing the specialization within the PPIM by splitting it into Big and Small PPIPs actually enables a significant number of new generalizations and features within the Big PPIP.

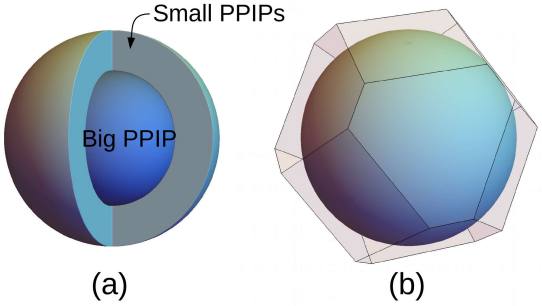


Fig. 4. An atom participating in interactions is at the center of the sphere. (a) Within the sphere, the innermost interactions are computed on the Big PPIP and the remainder are computed on the Small PPIP. (b) An enclosing polyhedron is used for filtering interactions in the L1 Match Unit.

Fig. 5 shows a block diagram of the Anton 3 PPIM. In addition to the PPIPs, new microarchitectural features include filtering, steering, and reduction logic to ensure that each interaction reaches the appropriate PPIP at the highest possible throughput and with good load balance. "Match Units" are responsible for comparing incoming atom positions with those in a "Stored Set" of atoms within PPIM memory, and for deciding which Stored Set atoms must interact with each incoming atom (i.e., whether the incoming atom is within R_{cut} of each Stored Set atom). To perform the matching operations efficiently at high throughput, the Match Units employ a hierarchical filtering structure.

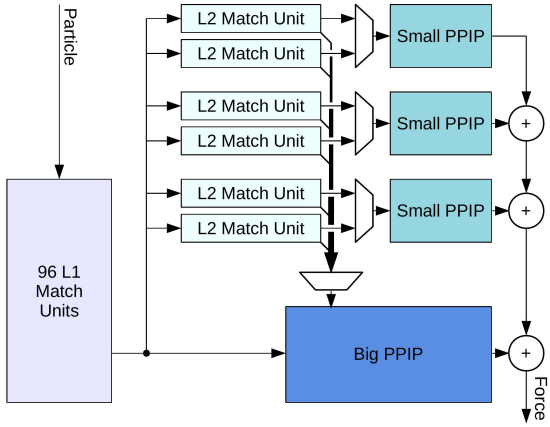


Fig. 5: Microarchitectural components of the PPIM.

The Level 1 (L1) Match Units begin the matching process by performing a conservative, low-precision distance check. A Euclidean distance check ($\Delta x^2 + \Delta y^2 + \Delta z^2 \leq R_{cut}^2$) defines a spherical region surrounding each Stored Set atom, but the L1 Match Unit instead uses the enclosing polyhedron shown in Equation 1 and Fig. 4b (a simpler polyhedron than described elsewhere [35]). The resulting Match Unit is highly area-efficient, enabling Anton 3 to include one L1 Match Unit per Stored Set atom and thereby provide a six-fold improvement in peak matching throughput per cycle over an Anton 2 PPIM.

$$|\Delta x| + |\Delta y| + |\Delta z| \leq \sqrt{3} R_{cut} \qquad (1)$$

$$\wedge\, |\Delta x| \leq R_{cut} \wedge |\Delta y| \leq R_{cut} \wedge |\Delta z| \leq R_{cut}$$

Each incoming atom that passes the L1 distance check is steered (using a simple load balancer) to a Level 2 (L2) Match Unit, which performs a high-precision Euclidean distance calculation while also checking whether the interaction is eligible to be computed on a Small PPIP. If the interaction is eligible for a Small PPIP (the more common case), the L2 Match Unit forwards the interaction to the nearest Small PPIP; otherwise, it is forwarded to the Big PPIP.

C. *Manhattan Method for Non-bonded Interactions*

Previous generations of Anton used the Neutral Territory (NT) method [15] to parallelize the computation of non-bonded, range-limited forces. The NT method is economical in its total network and compute bandwidth, but it has two key shortcomings: First, its bandwidth demand is not balanced across all dimensions of the 3D torus interconnect; and second, it requires each node to receive its entire Stored Set from other nodes before it can begin computing any forces, which adds latency to the time step and limits strong scaling. The *Manhattan Method*, conceived for Anton 3, balances the bandwidth demand across the torus dimensions and typically uses less total torus bandwidth than the NT method. Moreover, it allows each node to begin computing forces among atoms in its Home Box at the very beginning of the time step, without waiting to receive atoms

from other nodes. The key to these improvements is that, unlike NT, the Manhattan Method guarantees that each interaction is computed on a node that contains at least one of the interacting atoms within its Home Box.

Methods for parallelizing the calculation of interactions are defined by rules that specify where each interaction must be performed. The Manhattan Method requires that an interaction be calculated on the node whose Home Box atom has a larger *Manhattan distance* (the sum of the x, y, and z distance components) to the closest corner of the other node's Home Box, as shown in Fig. 6. If the Manhattan distances are equal, a simple and consistent tiebreaking rule is applied.

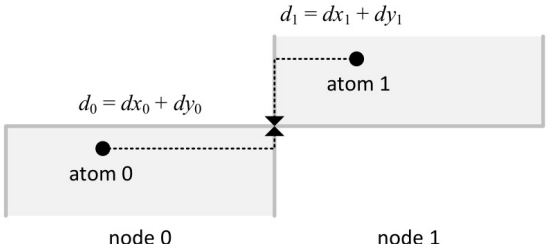


Fig. 6: 2D analog of Manhattan Method interaction rule. The interaction is performed on the node whose atom has a larger Manhattan distance d_i (the sum of the vertical and horizontal distance components) to the closest corner of the other node.

In Anton 3, the integration code running on all GCs sends, or "exports," every newly integrated position to all nodes on which the Manhattan Method might produce an interaction. Equivalently, each node receives, or "imports," a set of atoms that could interact on that node. Fig. 7 shows the import region for the Manhattan Method, with a realistic cutoff radius and per-node box size. The volume of the region, which is proportional to the number of atoms that must be imported onto any given node, is $3R_{cut}s^2 + 3R_{cut}^2 s + \sqrt{3}R_{cut}^3/2$, where s is the side length of the Home Box assigned to each node, and R_{cut} is the cutoff radius for non-bonded interactions.

Anton 3 includes hardware that checks the Manhattan Method's interaction criterion (Fig. 6) for each atom as it arrives on the node, and also at each PPIM; candidate interactions that do not meet the interaction criterion are filtered out. This enables some flexibility in the atom export routine, allowing atoms to be sent to a superset of the required destinations (when that is convenient from a hardware or software perspective) without affecting correctness.

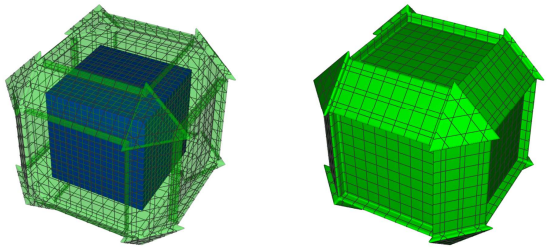


Fig. 7: Import region (green) for the Manhattan Method. The Home Box is in blue.

Fig. 8 shows an example of the dataflow for computing range-limited interactions with the Manhattan Method. At the beginning of a time step (a), each Core Tile holds a subset of the atom positions computed during the previous time step. These positions are broadcast (b) over the 2D mesh network to the Stored Sets of all PPIMs in the column and, concurrently, to the Edge Tiles. The Edge Tiles forward each atom to all nodes on which it might have an interaction. PPIM computation begins when all Stored Sets on a node have been loaded, which in the Manhattan Method requires only node-local communication. As atoms arrive at the Edge Tiles (c), they stream across the chip, interacting with Stored Set atoms in each PPIM. These interactions produce forces on the Stored Set atoms (which are accumulated within the PPIM) and on the streaming atoms (which accumulate along the Force Buses as the forces follow the streamed atoms across the chip). Forces on streamed atoms then enter the Edge Tile and return to the Core Tile (possibly on another node) that sent the atom position. Ultimately, the last forces on streamed atoms return to the Core Tiles (d). Stored Set forces are then unloaded from each PPIM and sent back to their originating Core Tiles (always on the same node), and integration begins.

Furthermore, Anton 3 supports a novel Hybrid Manhattan interaction method. Although the Manhattan Method is close to optimal for a wide range of simulation sizes, the less bandwidth-efficient Full Shell method [36] avoids the need to return forces to other nodes, thus reducing the time-step latency. Anton 3 can apply different interaction rules for different subsets of interactions—using Full Shell for interactions requiring more than one torus hop and Manhattan Method for the rest—further optimizing the balance between inter-node communication latency and bandwidth.

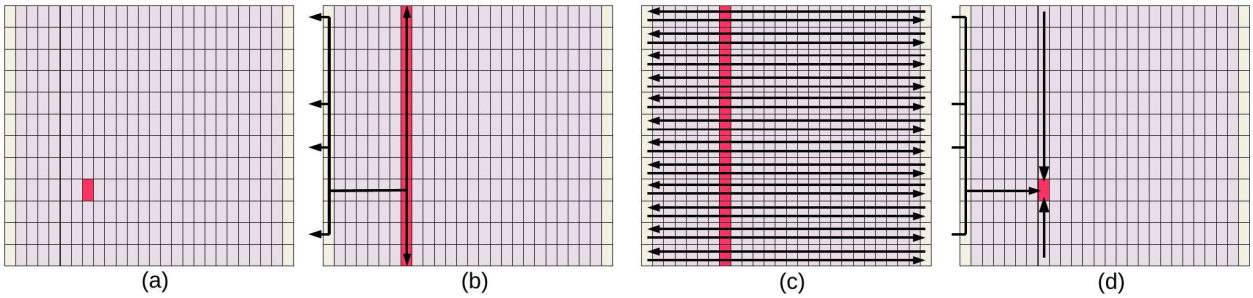


Fig 8: Abstract floorplan of the chip showing the tiled layout of Edge Tiles (yellow) and Core Tiles (purple) and example dataflow of the range-limited time step.

D. Specialized Bond Calculations

In previous generations of Anton, bond forces were computed on Geometry Cores (GCs). In Anton 3, 90% of bond forces are computed on the new Bond Calculator (BC). This increases performance by reducing the fraction of the chip area allocated to GCs in favor of area for higher-throughput, specialized pipelines. The smaller ratio of GCs to PPIPs (1:4 in Anton 3 vs. 1:2.4 in Anton 2) reflects this shift. At the same time, the BC reduces the energy cost of a typical simulation time step by about 20% (versus computing all bonds on the GCs), which is an important contribution to maintaining a reasonable power envelope for Anton 3. The BC computes interactions at a throughput that would otherwise heavily occupy two GCs, yet requires an area only one-third that of a single GC. Thanks to the work offloaded to the BC, the GCs are freed to perform other computations without impacting the critical path; this additional headroom helps to avoid hitting Amdahl's Law bottlenecks and allows for the addition of future software features.

As is the case with the Small PPIP, the BC does not attempt to compute all types of interactions, just the most common and numerically well-behaved interactions. These include most of the stretch, angle, and torsion forces common to the CHARMM [37] and AMBER [38] force fields. For each type of bond, the force is expressed as a function of a scalar internal coordinate (e.g., a bond length or angle) computed from the positions of the atoms participating in the bond. A GC sends these atom positions to the BC to be saved in a small cache, since an atom may participate in multiple bond terms. Subsequently, the GC sends the BC commands specifying the bond terms to compute, upon which the BC retrieves the corresponding atom positions from the cache and calculates the appropriate internal coordinate and thus, the bond force. The resulting forces on each of the bond's atoms are accumulated in the BC's local cache and sent back to memory only once per atom, when computation of all that atom's bond terms is complete.

VI. How Performance Was Measured

Performance was measured on real Anton 3 machines of up to 512 nodes.

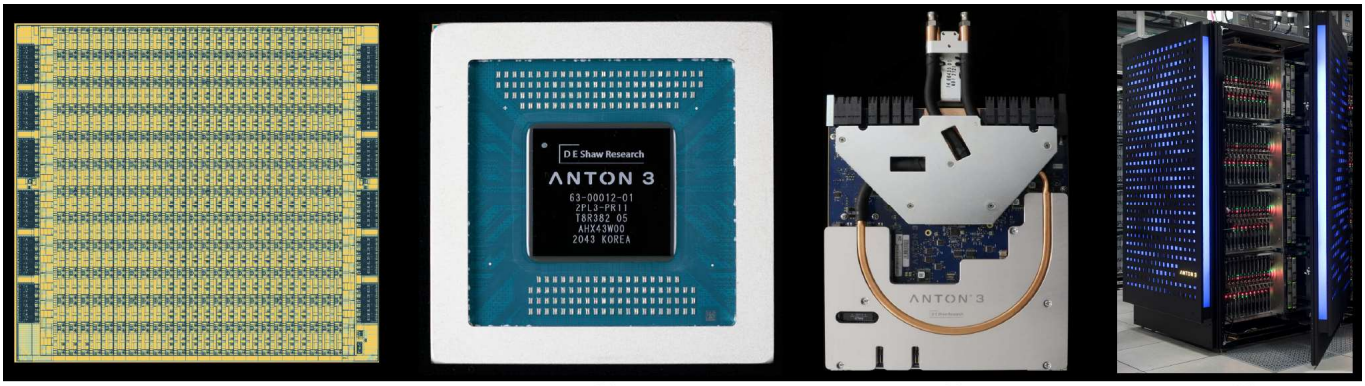
A. An Anton 3 System

Anton 3 chips (Fig. 9a,b) are mounted individually on water-cooled node boards, as shown in Fig. 9c. Power, cooling, and network connections at the back of each node board are "blind-mate" so that node boards can be inserted or removed quickly from the front of the rack. In addition to its 3D torus connections, each chip also connects to a low-power host processor that provides node control, management, and two external Gigabit Ethernet interfaces for data I/O. Each custom-designed rack in Fig. 9d holds 128 nodes, organized as four backplanes with 32 node boards each. Multiple backplane designs allow different torus sizes ranging from 2 × 2 × 2 (8 nodes) up to 8 × 8 × 8 (512 nodes). Up to five compute racks share one cooling rack (not shown), which contains pumps and a water-water heat exchanger.

Table I compares Anton 3 with earlier generations of Anton. A slightly larger die area and a new semiconductor process provide a 16-fold increase in the number of transistors over Anton 2. Major architectural changes, including those described in Section V, were crucial to extracting the maximum performance benefit from those additional transistors.

TABLE I. KEY DESIGN STATISTICS FOR THE ANTON FAMILY OF CHIPS. PPIPS AND GENERAL-PURPOSE CORES DIFFER ACROSS GENERATIONS NOT ONLY IN NUMBER BUT ALSO IN ARCHITECTURE AND CAPABILITIES.

	Anton 1	Anton 2	Anton 3
Power-on Year	2008	2013	2020
Semiconductor Process Node (nm)	90	40	7
Die Area (mm²)	305	408	451
# Transistors (B)	0.2	2	32
PPIP Clock Speed (GHz)	0.970	1.65	2.8
Non-PPIP Clock Speed (GHz)	0.485	1.65	2.8
# General-Purpose Cores	13	66	528
# PPIPs	32	152	2,112
# Bond Calculators	0	0	264
Typical Capacity (atoms per chip)	350	5,500	110,000
Typical Chip Power (W)	< 50	190	360
Total SERDES Bandwidth (GB/s)	76	336	696
Minimum Inter-Node Latency (ns)	162	99	55



(a) (b) (c) (d)

Fig. 9: (a) Anton 3 die, (b) packaged chip, (c) node board containing one chip, and (d) front view of a compute rack.

B. Software Configuration

Anton 3 supports a variety of MD force fields and simulation options, offering a spectrum of tradeoffs between fidelity and simulation speed. We configured our simulations as shown in Table II. These settings provide sufficient accuracy for most purposes and are generally similar to those commonly used for all-atom MD simulations on general-purpose computers.

Chemical systems [39] were selected from standard benchmark sets [27][40] and other sources to cover a wide range of chemical system sizes, from 24,000 atoms for a small standard benchmark (the DHFR protein) to 72 million atoms for an HIV-1 viral capsid. The STMV $5 \times 2 \times 1$ and STMV $5 \times 2 \times 2$ benchmarks (with 11 million and 21 million atoms, respectively) each comprise a spatial matrix of identical STM viruses. The selected benchmarks are representative of a much larger set of systems we have run on Anton 3, but that have been omitted here for brevity.

TABLE II. PARAMETERS CHOSEN FOR ANTON SIMULATIONS.

Force Field	Fixed charge
Solvents	Explicit water, rigid Reich [12] constraints
Range-limited Cutoff	9 ± 1 Å (depending on chem. system)
Distant Interactions	Particle-Mesh *u*-series [9], b = 2
Integrator	RESPA [10]
Time Step	2.5 fs (near), 7.5 fs (distant)
Thermostat	Applied every 48 time steps
Barostat	None

Generally, we found the highest single-GPU performance on an NVIDIA A100 SXM4 GPU with a dual-socket AMD EPYC 7F52 host under Desmond/GPU version 1.9.1.9 configured as recommended in the latest documentation. We refer to this platform as "Desmond A100." We have also reported slightly higher performance for DHFR on the NVIDIA A6000 GPU [20].

C. Measurements

Reported performance is computed using timestamps in the application logs. These times include I/O to periodically write out the simulation data, but exclude the initial application startup time in order to measure the steady-state sustained performance.

Power measurement is platform-dependent: For the Desmond A100 configuration, GPU power was measured with the nvidia-smi utility; on Anton 3, power was measured at the output of the regulators feeding the chip. Both measurements ignore some conversion losses and the power consumed in the host system. This arrangement favors the GPU when reporting power consumption because its host server requires far more power than Anton 3's host processor, which has a typical power of just 12 watts.

D. Reproducibility and Correctness

All Anton results are bitwise reproducible for a given set of input files and software version, even when re-run on machines of different sizes. This reproducibility is not only scientifically important, but also helps ensure the integrity of all simulation outputs. We continuously perform validation runs that select small random samples of simulations to re-run on each deployed machine and verify that such re-runs produce results identical to the original run. All released software versions are preserved over the lifetime of the machine, enabling reproduction of any data ever generated on the machine.

VII. PERFORMANCE RESULTS

Anton 3 sets new records in time-to-solution, per-node throughput, and energy efficiency. Improvements in any of these dimensions result in significant increases in scientific productivity of a machine's users, but the improvement in time-to-solution is especially noteworthy because absolute reductions in that metric require exceptional strong scaling.

A. Time-to-Solution

Table III shows the rates at which simulations run on Anton 3 and a variety of other contemporary machines, including those that have achieved the best-available non-Anton results we have found in the literature. The table illustrates the enormous time-to-solution advantage of Anton 3 over all other machines (including a 512-node Anton 2 machine) across a wide range of chemical systems. Even on a 64-node Anton 3, chemical systems of ten thousand to over two million atoms run at least 100-fold faster than on any non-Anton machine. Systems of up to ~21 million atoms run at least 100 times faster on a 512-node Anton 3 than on any other machine.

TABLE III: SIMULATION RATES (MICROSECONDS PER DAY) FOR ANTON 3, ANTON 2, AND THE FASTEST NON-ANTON MACHINES ON BENCHMARK SYSTEMS.

	DHFR	ApoA1	ATPase	STMV	Ribosome	STMV 5×2×1	STMV 5×2×2	HIV-1 Capsid
# atoms	24K	92K	328K	1,067K	2,181K	10,666K	21,333K	72,404K
Anton 3 512-node (µs/day)	—	—	166.9	121.1	93.2	22.4	16.0	1.9
Anton 3 64-node (µs/day)	212.2	152.4	90.6	42.3	25.4	—	—	—
Anton 2 512-node (µs/day)	87.2	63.7	34.5	13.9	5.0	—	—	—
Non-Anton (µs/day)	1.70 (*a*) 1.46 (*b*) 0.63 (*c*)	0.57 (*b*) 1 (*d*)	0.20 (*b*)	0.064 (*b*) 0.055 (*e*)	0.204 (*f*) 0.031 (*b*)	0.12 (*g*)	0.12 (*h*)	0.008 (*i*)

(*a*) Desmond/GPU on single Quadro RTX A6000 GPU [20]; (*b*) Desmond/GPU on single A100 SXM4 GPU [20]; (*c*) custom implementation on Stratix 10 FPGA [33]; (*d*) MDGRAPE-4A simulating SARS-CoV-2 M^{pro} (at 99K atoms, similarly sized to ApoA1) [32]; (*e*) GROMACS 2020 on 8 V100 GPUs [21]; (*f*) GROMACS 4.6 on 512 nodes of HYDRA cluster [41]; (*g*) NAMD on 256 nodes of Summit [42]; (*h*) NAMD on 1024 nodes of Summit [42]; (*i*) a smaller 64,424K-atom version of the capsid system simulated with NAMD on 3880 nodes of Titan [43].

Biochemical systems exhibit interesting events at different rates over many timescales, so there is no canonical length for an MD simulation. Our experience with previous Anton machines shows that biochemically important motions in systems relevant to drug discovery often only appear after simulating for several microseconds, and statistics from Anton 2 usage show that the median simulation length is approximately 20 µs (with a distribution ranging from sub-microsecond to multi-millisecond lengths). A single project generally requires many simulations, and an entire drug discovery campaign can involve thousands.

Fig. 10 shows the simulation speed on various sizes of Anton 3 machines with respect to problem size (chemical system in number of atoms) as well as for a 512-node Anton 2 machine and other machines from Table III. The green region in the plot highlights the vast range of machine sizes and chemical systems for which a 20-µs simulation can be run within one week. A 20-µs simulation of a 2.2-million-atom ribosome runs in about five hours on a 512-node Anton 3, while the fastest general-purpose supercomputers would take about 100 days to perform the same simulation. For this simulation, a 512-node Anton 3 is roughly 19 times faster than Anton 2, and 460 times faster than any general-purpose machine. A 20-µs simulation of a 21-million-atom system (STMV 5 × 2 × 2) runs in under two days on a 512-node Anton 3 machine—about 130 times faster than is possible on any other machine.

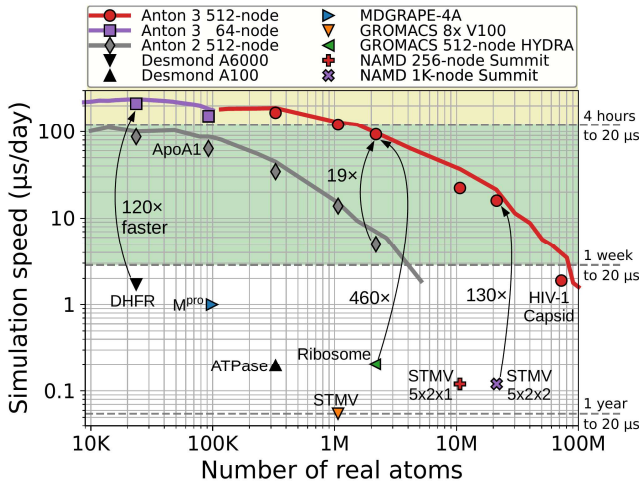


Fig. 10: Simulation speed for various benchmarks and machines. Annotations in the right margin show the wall-clock time required to complete a 20-µs simulation; the green region contains simulations that reach 20 µs in under a week, and the yellow region contains simulations that run 20 µs in under 4 hours. The plotted lines show a synthetic water-only benchmark at various atom counts, indicating the general performance scaling trend on Anton systems. The markers represent real simulations of specific chemical systems.

The yellow region of Fig. 10 highlights the combination of machine sizes and chemical system sizes for which 20-µs simulations can be run in under four hours, offering the best possible time-to-solution. We expect simulations with such fast turnaround times to enhance scientific productivity by enabling multiple hypothesis-test-refine iterations in a single day. No other machine achieves this time-to-solution for chemical systems of any size. A 512-node Anton 3 is capable of four-hour turnaround times on systems all the way up to a million atoms.

Anton 3's high performance across such a wide range of problem and machine sizes is largely due to its excellent strong scaling (performance on a fixed chemical system size). Even for a small 24,000-atom benchmark (DHFR), where strong scaling is most difficult, a 64-node Anton 3 achieves 212 µs/day. This is more than double the previous speed record, held by a 512-node Anton 2 and more than 120 times faster than any general-purpose machine, a significant achievement in this extreme scaling scenario in which the 64-node Anton 3 machine has less than a single atom per GC. On this widely reported benchmark, general-purpose machines have yet to achieve 2 µs/day (reaching 1.7 µs/day on a single A6000 GPU with no communication overhead; multi-node general-purpose machines have not reported any results above 0.5 µs/day).

B. Throughput

Scientists can optimize the aggregate simulation time for multiple concurrent simulations (while still achieving exceptional time-to-solution) by using machines for which their problem sizes lie within the green region of Fig. 10. The dramatic gains in throughput (i.e., number of simulations per unit time) delivered by Anton 3 become especially clear when migrating simulations from Anton 2: A 64-node Anton 3 is strictly superior to a 512-node Anton 2, achieving higher speed over the entire range of chemical system sizes supported by the latter, and running many that would not even fit within the capacity of an Anton 2. By moving, for example, an ensemble of ribosome simulations from a 512-node Anton 2 machine to the same total number of Anton 3 nodes configured as eight 64-node machines, a scientist would obtain a 40-fold improvement in throughput *as well as* a 5-fold improvement in time-to-solution on each of those simulations. The high per-node atom capacity of Anton 3 affords great flexibility in making these throughput/time tradeoffs on a project-by-project basis. For a historical perspective, it is also worth noting that a single Anton 3 node outperforms a 512-node Anton 1 across the former's entire range of supported chemical system sizes.

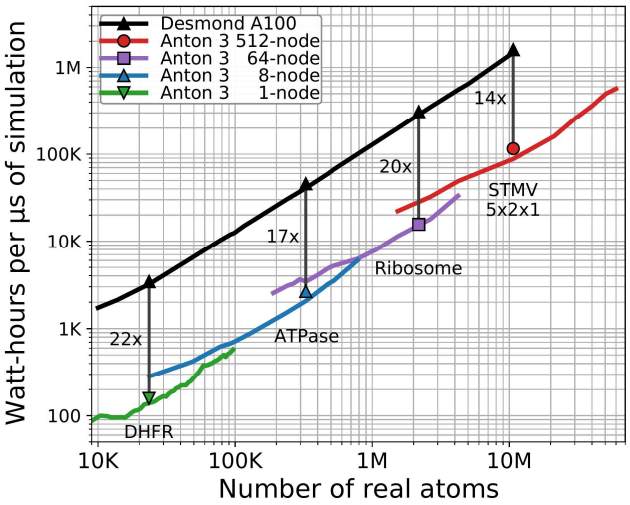


Fig. 11. Energy consumption per microsecond of simulation.

C. Energy Efficiency

Energy consumption is of increasing concern in high-performance computing, as it is a dominant component of the cost of operation of any supercomputer, as well a social concern with respect to climate change. Among commodity hardware, GPUs offer the highest energy efficiency for MD [44]. Fig. 11 shows that energy efficiency of MD on Anton 3 is at least ten times better than on the fastest contemporary GPU, even while Anton 3 offers a much faster time-to-solution (Section VII-A). Anton 3 thus simultaneously achieves much faster time-to-solution, much higher throughput, and much better energy efficiency than general-purpose machines.

VIII. Implications

Before Anton 3, there was no practical way for all-atom MD simulations of chemical systems above a few million atoms to reach multi-microsecond timescales; Anton 3 makes that possible for systems of more than 50 million atoms, which will provide insight into entirely new areas of biochemistry. Furthermore, the order of magnitude improvement in performance across most chemical systems under three million atoms offers extraordinarily deeper visibility into system sizes already supported by Anton 2. The accessible timescale can be extended even further by using Anton to apply enhanced sampling techniques [45][46] and inform coarse-grained models [47]. Finally, by offering increased performance while using far fewer nodes and overall energy, far more simulations can be run concurrently (given a fixed number of nodes).

Most existing force fields (e.g., CHARMM and AMBER) are optimized to provide reasonable accuracy at low computational cost in order to reach even modest timescales with conventional hardware resources. The greatly increased throughput and faster turnaround offered by Anton 3 machines will enable further force-field refinement through the comparison of large numbers of long-timescale simulations with experimental data [48][49][50]. Furthermore, increasing the accuracy of force fields requires much more computation in order to model physical effects such as polarization of atomic charge distributions [51]. Anton 3 will enable the use of more accurate force fields while still retaining access to long timescales of biological interest.

It is often desirable to search a very large number (potentially billions) of drug-like molecules to find examples that bind strongly to a particular site on a protein, thus influencing the protein's behavior in a desired manner. Computationally, these "virtual screens" often use approximate "docking" methods that are considerably faster but less accurate than MD simulation. We are currently experimenting with software that uses Anton 3's specialized hardware to perform docking calculations at far greater speed than general-purpose machines. Combining docking-based virtual screening on Anton 3 with detailed MD simulations of the best screening candidates on Anton 3 may further improve the latency, throughput, and quality-of-result of the drug discovery process.

The Anton family continues to offer a case study in the value of clean-sheet design, reusing few hardware or software implementation details from the preceding generations to optimize each generation for the rapidly changing constraints and scaling opportunities that semiconductor technologies have presented. Designs for effective next-generation specialized machines will almost certainly benefit from a similar approach in which the entire architecture—from algorithms to transistors—is designed from the ground up to extract large improvements in capabilities, time-to-solution, per-node throughput and energy efficiency from a changing technology landscape.

Acknowledgments

We thank Morten Jensen for insights into ion channels and for Fig. 1; our Chemistry team, and in particular Stefano Piana, Albert Pan, and Vish Jogini, for chemistry validation; Steve Sandler for modeling power-supply performance; our Force Field team, and in particular Brent Gregersen, for helpful discussions about force fields; Nathan Olla, Sanjeev Pandey, Clayton Falzone, Eric Radman, and Amit Kshirsagar for systems infrastructure support; Julie Driansky and our Operations team, Želimir Galjanić and our Strategic Growth team, and Segnon Tiewul and our Admin team for operational assistance; Eric Martens and Berkman Frank for editorial assistance; and the entire DESRES team, past and present, for countless other forms of input and support.

[b] Current affiliations of indicated authors, who contributed to this work while employed by D. E. Shaw Research: Peter J. Adams: Tesla; Jhanvi Bhatt: Morgan Stanley; Ron O. Dror: Stanford University; Maria Gorlatova: Duke University; J.P. Grossman: Citadel Securities; William Hasenplaugh: Citadel Securities; Mollie M. Kirk: Jane Street; Richard McGowen: Waymo; Mohamed H. Nasr: Tesla; Carl Schwink: Draper; Naseer Siddique: Waymo; Jochen Spengler: Cerebras Systems; Raymond Tabladillo: Tabworks; Reinhard Tartler: Bloomberg; Brian Towles: Google; William Vick: Advanced Micro Devices; John M. Williams: Independent Contractor. [c] Our dear colleagues, Robert M. Dirks and John K. Salmon, passed away on February 3, 2015, and October 27, 2021, respectively.

References

[1] S. A. Hollingsworth and R. O. Dror, "Molecular dynamics simulation for all," *Neuron,* vol. 99 (6), 2018, 1129–1143.

[2] T. Schlick et al., "Biomolecular modeling and simulation: A prospering multidisciplinary field," *Annual Review of Biophysics*, vol. 50, 2021.

[3] D. E. Shaw et al., "Anton, a special-purpose machine for molecular dynamics simulation," in *Proceedings of the 34th Annual International Symposium on Computer Architecture*, 2007, 1–12.

[4] D. E. Shaw et al., "Millisecond-scale molecular dynamics simulations on Anton," in *SC '09: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, 2009, 1–11.

[5] D. E. Shaw et al., "Anton 2: Raising the bar for performance and programmability in a special-purpose molecular dynamics supercomputer," in *SC'14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 2014, 41–53.

[6] National Academies of Sciences, Engineering, and Medicine, "Report of the committee on proposal evaluation for allocation of supercomputing time for the study of molecular dynamics," Washington, DC: The National Academies Press, vols. 1–11, 2010–2020.

[7] T. Darden et al., "Particle Mesh Ewald: An N log (N) method for Ewald sums in large systems," *The Journal of Chemical Physics*, vol. 98 (12), 10,089–10,092, 1993.

[8] Y. Shan et al., "Gaussian Split Ewald: A fast Ewald mesh method for molecular simulation," *The Journal of Chemical Physics*, vol. 122 (5), 054101, 2005.

[9] C. Predescu et al., "The *u*-series: A separable decomposition for electrostatics computation with improved accuracy," *The Journal of Chemical Physics*, vol. 152 (8), 2020, 084113.

[10] M. Tuckerman et al., "Reversible multiple time scale molecular dynamics," *The Journal of Chemical Physics*, vol. 97 (3), 1992, 1990–2001.

[11] J.-P. Ryckaert et al., "Numerical integration of the Cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes," *Journal of Computational Physics*, vol. 23 (3), 1977, 327–341.

[12] R. D. Skeel and S. Reich, "Corrected potential energy functions for constrained molecular dynamics," *The European Physical Journal Special Topics*, vol. 200 (1), 55–72, 2011.

[13] K. A. Feenstra et al., "Improving efficiency of large time-scale molecular dynamics simulations of hydrogen-rich systems," *Journal of Computational Chemistry*, vol. 20 (8), 1999, 786–798.

[14] S. Piana, K. Lindorff-Larsen, and D. E. Shaw, "Atomic-level description of ubiquitin folding," *Proceedings of the National Academy of Sciences*, vol. 110 (15), 2013, 5915–5920,.

[15] D. E. Shaw, "A fast, scalable method for the parallel evaluation of distance-limited pairwise particle interactions," *Journal of Computational Chemistry*, vol. 26 (16), 2005, 1803–1803,.

[16] K. Fatahalian and M. Houston, "A closer look at GPUs," *Communications of the ACM*, vol. 51 (10), 2008.

[17] M. J. Harvey et al., "ACEMD: Accelerating biomolecular dynamics in the microsecond time scale," *Journal of Chemical Theory and Computation*, vol. 5 (6), 2009, 1632–1639.

[18] R. Salomon-Ferrer et al. "Routine microsecond molecular dynamics simulations with AMBER on GPUs. 2. Explicit Solvent Particle Mesh Ewald," *Journal of Chemical Theory and Computation*, vol. 9 (9), 2013, 3878–3888.

[19] B. R. Brooks et al., "CHARMM: The biomolecular simulation program." *Journal of Computational Chemistry*, vol. 30 (10), 2009, 1545–614.

[20] M. Bergdorf et al, "Desmond/GPU Performance as of April 2021", DESRES/TR--2021-01, [Online: April 2021] https:// deshawresearch.com/publications.html#technicalreports.

[21] S. Páll et al., "Heterogeneous parallelization and acceleration of molecular dynamics simulations in GROMACS," *The Journal of Chemical Physics*, vol. 153 (13), 2020.

[22] J. A. Anderson et al., "HOOMD-blue: A Python package for high-performance molecular dynamics and hard particle Monte Carlo simulations," *Computational Materials Science*, vol. 173, 2020, 109363.

[23] S. Plimpton, "Fast parallel algorithms for short-range molecular dynamics," *Journal of Computational Physics*, vol. 117 (1), 1995, 1–19. https://lammps.sandia.gov/index.html

[24] J. C. Phillips et al., "Scalable molecular dynamics on CPU and GPU architectures with NAMD," *The Journal of Chemical Physics*, vol. 153 (4), 2020, 044130.

[25] P. Eastman et al., "OpenMM 7: Rapid development of high performance algorithms for molecular dynamics." *PLOS Computational Biology*, vol. 13 (7), e1005659, 2017.

[26] About OpenMM. [Online: March 2021]. http://openmm.org/about.html

[27] Amber GPU benchmarks. [Online: March 2021]. https://ambermd.org/gpus16/benchmarks.htm

[28] C. Kobayashi et al., "GENESIS 1.1: A hybrid-parallel molecular dynamics simulator with enhanced sampling algorithms on multiple computational platforms." *Journal of Computational Chemistry*, vol. 38, *2017,* 2193–2206.

[29] J. Glaser et al., "Strong scaling of general-purpose molecular dynamics simulations on GPUs," *Computer Physics Communications*, vol. 192, 2015, 97–107.

[30] L. Casalino et al., "AI-driven multiscale simulations illuminate mechanisms of SARS-CoV-2 spike dynamics," *International Journal of High Performance Computing Applications*, 2021.

[31] I. Ohmura, G. et al., "MDGRAPE-4: a special-purpose computer system for molecular dynamics simulations," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 372 (2021), 20130387, 2014.

[32] "RIKEN Team Use Supercomputer to Explore the Molecular Dynamics of the New Coronavirus," HPCwire announcement, [Online: March 2020]. https://www.hpcwire.com/off-the-wire/riken-team-use-supercomputer-to-explore-the-molecular-dynamics-of-the-new-coronavirus/

[33] C. Yang et al., "Fully integrated FPGA molecular dynamics simulations," In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (SC '19), Article 67, 2019, 1–31.

[34] M. Schaffner and L. Benini, "On the feasibility of FPGA acceleration of molecular dynamics simulations," https://arxiv.org/abs/1808.04201, 2018.

[35] M. Chiu and M. C. Herbordt, "Efficient particle-pair filtering for acceleration of molecular dynamics simulation," in *2009 International Conference on Field Programmable Logic and Application,*. IEEE, 2009, 345–352.

[36] K. J. Bowers et al., "Overview of neutral territory methods for the parallel evaluation of pairwise particle interactions," in *Journal of Physics: Conference Series*, vol. 16 (1), IOP Publishing, 2005, 041.

[37] J. Huang et al., "CHARMM36m: An improved force field for folded and intrinsically disordered proteins," *Nature Methods*, 14 (1), 2017, 71–73.

[38] C. Tian et al. "ff19SB: Amino-Acid-Specific Protein Backbone Parameters Trained against Quantum Mechanics Energy Surfaces in Solution." *Journal of Chemical Theory and Computation* vol. 16 (1), 2020, 528–552.

[39] Desmond/GPU benchmark systems. [Online: August 2021]. https://www.deshawresearch.com/downloads/download_trajectory_benchmark2021.cgi/

[40] NAMD utilities. [Online: March 2021]. http://www.ks.uiuc.edu/Research/namd/utilities/

[41] S. Páll et al., "Tackling exascale software challenges in molecular dynamics simulations with GROMACS," In: Markidis S., Laure E. (eds) Solving Software Challenges for Exascale. EASC 2014. Lecture Notes in Computer Science, vol. 8759, 2015.

[42] NAMD scaling on Summit. [Online: May 2018]. http://www.ks.uiuc.edu/Research/namd/benchmarks/

[43] J. R. Perilla and K. Schulten, "Physical properties of the HIV-1 capsid from all-atom molecular dynamics simulations," *Nature Communications*, vol. 8 (15959), 2017.

[44] C. Kutzner et al., "More bang for your buck: Improved use of GPU nodes for GROMACS 2018," *Journal of Computational Chemistry*, vol. 40 (27), 2418–2431, 2019.

[45] Pan, Albert C et al. "Demonstrating an Order-of-Magnitude Sampling Enhancement in Molecular Dynamics Simulations of Complex Protein Systems." *Journal of Chemical Theory and Computation*, vol. 12 (3), 2016, 1360–1367.

[46] Pan, Albert C et al. "Atomic-level characterization of protein-protein association." *Proceedings of the National Academy of Sciences*, vol. 116 (10) 2019, 4244–4249.

[47] A. Yu et al., A multiscale coarse-grained model of the SARS-CoV-2 virion. *Biophysical Journal*, 120 (6), 2021, 1097–1104.

[48] K. Lindorff-Larsen et al., "Systematic validation of protein force fields against experimental data," *PLOS ONE*, vol. 7 (2), 2012, e32131.

[49] S. Piana et al., "Development of a force field for the simulation of single-chain proteins and protein-protein complexes." *Journal of Chemical Theory and Computation,* vol. 16 (4), 2020, 2494–2507.

[50] D. Tan et al., "RNA force field with accuracy comparable to state-of-the-art protein force fields." *Proceedings of the National Academy of Sciences*, vol. 115 (7), 2018, E1346–E1355.

[51] F.-Y. Lin et al., "Further optimization and validation of the classical Drude polarizable protein force field," *Journal of Chemical Theory and Computation*, 16 (5), 2020, 3221–3239