ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

COMPUTATIONAL ACCELERATOR PHYSICS: ON THE ROAD TO EXASCALE

R. D. Ryne*, Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA

Abstract

The first conference in what would become the ICAP series was held in 1988. At that time the most powerful computer in the world was a Cray YMP with 8 processors and a peak performance of 2 gigaflops. Today the fastest computer in the world has more than 2 million cores and a theoretical peak performance of nearly 200 petaflops. Compared to 1988, performance has increased by a factor of 100 million, accompanied by huge advances in memory, networking, big data management and analytics. By the time of the next ICAP in 2021 we will be at the dawn of the Exascale era. In this talk I will describe the advances in Computational Accelerator Physics that brought us to this point and describe what to expect in regard to High Performance Computing in the future. This writeup as based on my presentation at ICAP'18 along with some additional comments that I did not include originally due to time constraints.

INTRODUCTION AND BRIEF HISTORY

The first conference in what would become the Computational Accelerator Physics series was held 30 years ago in San Diego, California in January 1988. At the time I was 28 years old. The meeting was called the Conference on Linear Accelerator and Beam Optics Codes [1]. I think there are three of us here now who were present for that meeting: Martin Berz, Herman Wollnik, and me. I'll describe the ICAP conference series in a moment, but first want to briefly address the the origins of the field of Computational Accelerator Physics. This summary is based on the paper, "Oh Camelot! A memoir of the MURA years," by F.T. Cole [2].

As is well known, Lawrence invented the first cyclotron in 1930 inspired by the work of Rolf Wideroe on resonance acceleration. In 1940 Donald Kerst built the first betatron, a 2 MeV electron machine. Soon after WWII Edwin McMillan was at Los Alamos waiting to return to Berkeley. According to Cole, McMillan told him that, in a single evening, he worked out the concepts for the sychrocyclotron and the synchrotron. Independently in the Soviet Union Vladimir Veksler did the same. Two proton synchrotrons were built in the early 1950's to go beyond a GeV, the Cosmotron at Brookhaven and the Bevatron at Berkeley.

Along with progress in circular accelerators there were also developments in linear accelerators. Luis Alvarez developed the first proton linac at Berkeley in 1948. Also, developments in radar during WWII led to high frequency, GHz power sources that Hanson and Panofsky used to develop electron linacs at Stanford.

A revolution in accelerator physics took place in 1952 with the invention of strong focusing by Courant, Snyder, and Livingston. As it turns out, Nick Christopholis had actually filed for a patent on strong focusing in 1950 and it was eventually granted in 1954. John Blewitt (BNL) applied alternating-gradient focusing to high intensity linacs. Also, the concept of Fixed Field Alternating Gradient (FFAG) was invented independently by multiple researchers, including Symon in 1954.

Strong focusing provided a totally new approach to high energy accelerators. A new lab, CERN, was founded after the war. Thanks to Lew Kowarski CERN acquired its first electronic computer in 1958. The CERN PS was commissioned in the Fall of 1959. The 30 GeV AGS at BNL began operation in 1960.

THE BEGINNING OF COMPUTATIONAL ACCELERATOR PHYSICS

So far I've described some key developments in accelerator physics through the 1950's. The 1950's also brings us to the first digital computations for accelerator modeling. While there was plenty of activity in the field, I would particularly like to mention the work of L. Jackson Laslett. Laslett was a pioneer in using digital computers for orbit calculations and for calculating electromagnetic fields. There are records of Laslett performing his simulations on a computer known as the ILLIAC I, a computer comprised of 2800 vacuum tubes. While working for the Midwestern Universities Research Association Laslett observed and analyzed sensitive dependence on initial conditions – what we now call chaos. He did this in the mid 1950's. His studies actually predate the work of Edward Lorenz who discovered chaos in weather simulations and whose 1962 paper launched chaos theory. Of course mathematicians going back to Poincare had predicted dynamical behavior that we now describe as chaotic dynamics.

I would also like to mention another important event of the 1950's involving scientific computing that included someone who would later become heavily involved in Computational Accelerator Physics. That event was the simulation of the Fermi-Pasta-Ulam (FPU) problem, and the person involved was Mary (Tsingou) Menzel [3]. Mary was the programmer for the FPU problem on the MANIAC computer at Los Alamos National Laboratory (LANL). I met her in the 1980's. By then she was a member of the Accelerator Technology Division at LANL and I was a graduate student who spent my summers there. I remember Mary telling me there were cans of water on top of the computer for cooling!

Along with computational developments, there were also key theoretical developments in the 1950s. Most notably, Kolmogorov published his original paper in 1954, which set the stage for the KAM Theorem. A key consequence,

* rdryne@lbl.gov

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

relevant to long-term dynamics in circular accelerators, is that, under sufficiently small perturbations of an integrable Hamiltonian system, there remains a set of initial conditions whose orbits are quasi-periodic. Let me also mention that Jim Ellison is in the audience, and Jim's former student Scott Dumas has published a wonderful book on KAM history [4].

As evidence of the growing appreciation of computing, let me mention that the 1967 issue of CERN Courier was devoted to "the electronic computer and its use at CERN" [5]. Since this conference is attended by a large number of programmers I thought it would be interesting to show this quote from that issue:

The designers of the early computers assumed that programming would be done by small groups of specialists, probably mathematicians, and that it would be undesirable to make the task too easy. For example, von Neumann and Goldstine, who in 1946 proposed what is essentially the modern computer, argued against built-in floating-point arithmetic: "The floating binary point represents an effort to render a thorough mathematical understanding of at least part of the problem unnecessary, and we feel that this is a step in a doubtful direction."

In 1972 a second issue of the CERN Courier was published that was devoted to computers [6]. The opening article was by Lew Kowarski and titled, "Computers: Why?" It is remarkable how prescient Kowarski was about how computers would be used in the future. In the article he states, "We are only beginning to discover and explore the new ways of acquiring scientific knowledge which have been opened by the advent of computers…" He then goes on to state eight modes of application: (1) numerical mathematics, (2) data processing, (3) symbolic calculations, (4) computer graphics, (5) simulation, (6) file management and retrieval, (7) pattern recognition, and (8) process control.

EARLY ACCELERATOR CODES

The preceding developments in digital computing and in accelerator theory and dynamical systems theory would lead to the topics that we address in these conference series. The code TRANSPORT came on the scene in the 1960's. It was original developed by Karl Brown at SLAC [7]. A second-order version was released around 1969. (For those of you familiar with the Berkeley Lab's Computational Research Division, the division head, David Brown, is Karl's son.) Dave Cary, at Fermilab, developed a third-order version TRANSPORT. Ed Heighway, at Los Alamos, developed a version called TRANSOPTR for design optimization.

A breakthrough in single-particle optics came with the invention of Lie Algebraic methods. In the USA this was led by Alex Dragt and his group at the University of Maryland [8] Alex was a originally a theorist in elementary particle physics. He later applied his skills to plasma physics, and with John Finn published the Dragt-Finn factorization theorem [9]. This shows how a Taylor series, as represented in a code like TRANSPORT, can be represented as a factored product of Lie transformations, as in a code like MaryLie. See Fig. 1.

Alex's involvement in Accelerator Physics came by accident. He was planning a sabbatical in the Plasma Physics Division at Los Alamos in 1978/79 when the division folded. Fortunately Richard Cooper suggested to Alex that he do his sabbatical in the Accelerator Theory group that he headed in the Accelerator Technology Division at Los Alamos. This launched Alex's involvement in Accelerator Physics. And the rest is history…

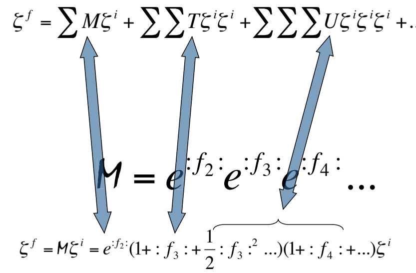

Figure 1: Correspondence between a map represented as a Taylor series and a map represented as a factored product of Lie transformations.

Alex and his student Etienne Forest published an article on the equations of motion for the matrix M and for the polynomials $f_3, f_4 \ldots$, in the Lie algebraic representation of the transfer map [10]. This opened the door to computing transfer maps for realistic beamline elements, i.e., for elements with fringe fields. The application of this became a portion of my Ph.D. thesis, known as the "genmap" capability in MaryLie. Eventually it was used to model realistic solenoids, dipoles, quadrupoles, and RF cavities [11].

Alex Dragt is retired but still active in the field. For those who would like to learn about Lie Algebraic methods, Alex has written a more than 2500 page book, that is freely available, "Lie Methods for Nonlinear Dynamics with Applications to Accelerator Physics" [12].

Though this talk is mainly about beam dynamics it is worth mentioning that the Superfish code, developed by Klaus Halbach and Ron Holsinger, was released in 1976 [13]. Later the Poisson and Superfish codes were maintained and developed by the Los Alamos Accelerator Code Group (LAACG). Though the Poisson and Superfish codes are only 2D they are still widely used for the early stages of accelerator design. The codes PARMILA, PARMELA, PARMTEQ, TRACE, and TRACE3D were all developed at Los Alamos.

In Europe, the first version of the code MAD was developed in the early 1980's [14]. This was led by F. Christoph Iselin along with Jim Niederer and Eberhard Keil. Originally a TRANSPORT-like code, Christoph eventually put large portions of MaryLie inside MAD. It's also worth mentioning that people like Karl Brown, Dave Cary, Christoph Iselin, and others, led an effort to develop a common input format that many of the major beam dynamics codes now use.

So far I have mentioned map-based codes like TRANSPORT, MaryLie, and MAD. Starting in the 1980's a different

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

approach emerged, based on direct numerical integration of the equations of motion by a symplectic method. This involved many people, but in accelerator physics it began with the third-order integrator of Ruth [15]. Later Forest and Ruth derived a fourth-order integrator using Lie methods [16]. Yoshida showed how to obtain an integrator of order 2n+2 by combining integrators of order 2n [17]. The work of Yoshida was later extended by Forest *et al.* [18].

Another major development was the application of Differential Algebraic techniques and automatic differentiation to beam dynamics by Martin Berz [19]. This opened the door to performing Taylor series calculations to arbitrary order, as implemented in the code COSY-INFINITY [20]. Along with all these advances came the development of normal form techniques [21] that have been critical to understanding global properties of periodic transport systems and designing these systems.

The mention of COSY-INFINITY brings me back to the first Computational Accelerator Conference in 1988. The proceedings contain papers related to many of the codes above. And it is interesting to note that a paper by Berz mentions a code "under the tentative name COSY INFINITY."

As a sign of how much things have changed since 1988, consider this quote from a paper in the proceedings: "The problem shown required 22 seconds on the IBM 3080 and 23 minutes on a machine with 8 MHz clock… The PC had the Intel 80287 Math co-processor and 1.1 Mbyte storage…"

As a sign of the promise of the future, consider this quote from a paper by Ed Heigthway: "…the beam transport designer's world is richer and probably evolving faster than at any time since Karl Brown first put finger to keypunch." At this point in my presentation I felt obliged to show a picture of a keypunch machine because I thought that some people in the audience might not know what I was talking about.

Let me mention one more thing before leaving that 1988 conference: The proceedings say nothing about parallel computing. But that was about to change.

1990's: PARALLEL COMPUTING ENTERS

The next meeting in the series was in 1990, hosted by Los Alamos. It was called the Conference on Computer Codes and the Linear Accelerator Community [22]. I counted five papers in the proceedings that mentioned parallel computing, although some of those described compatibility with parallel processing, not that they were actually doing it. One that I will mention specifically is, "Wakefield Calculation on Parallel Computers," by Paul Schoessow. He mentions finite difference codes that run on an Alliant FX/8 and on a Connection Machine CM-2. This is the earliest paper I'm aware of on massively parallel accelerator modeling.

The next conference, called CAP'93, was held in 1993 in Pleasanton, California [23]. It was co-organized by me and Susarla Murty. This was the last CAP conference before the Superconducting Supercollider (SSC) was cancelled, and there were several talks from people associated with the SSC.

The 1990's saw massively parallel computing emerge as a major new paradigm. My involvement came at the Advanced Computing Laboratory at Los Alamos. There, with Salman Habib and other colleagues, we developed early parallel beam dynamics codes and, eventually, the first version of the IMPACT code [24]. To a large extent this was motivated by a desire to use a large number of macroparticles to simulate very low density beam halos. It also opened the door to performing practical 3D space-charge calculations. We did this mainly on a computer by Thinking Machines Corporation called the Connection Machine 5. At first we used a technique from computational cosmology by Ferrell and Bertschinger to compute space-charge effects. But eventually we we adopted parallel particle-in-cell techniques of Paulette Liewer, Victor Decyk, and others [25].

During the 1980's when I spent my summers at Los Alamos I remember thinking that the space-charge code developers (who were interested in high intensity linacs) and the single-particle optics modelers (who were interested in aberrations, dynamic aperture, fringe fields, etc.) did not interact much. By the mid-1990's my experience with high-order optics and parallel particle-in-cell methods, along with symplectic integrators, led me to introduce split-operator methods as a means to combine the best of both worlds [26, 27]. See Fig. 2.

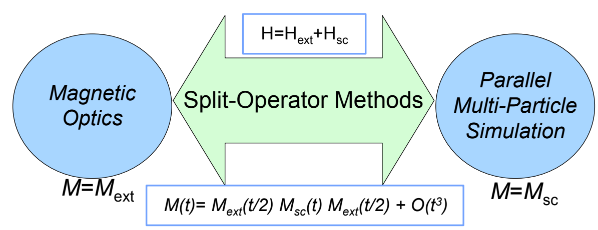

Figure 2: Split-operator method for combining high-order optics with space-charge.

The next conference, called CAP'96, was held in Williamsburg in 1996 [28]. At about this time (1996) NERSC moved to Lawrence Berkeley National Lab. Also, computer systems were shifting away from vector machines to massively parallel machines. I remember being involved in an email exchance about what the next big computer should be at NERSC. A decision was made that it should be a massively parallel Cray T3E. It was called mcurie.

By this time parallel computing was becoming a major activity in the accelerator physics community. In the USA, in 1997, the US Department of Energy launched the DOE Grand Challenge in Computational Accelerator Physics. The first Terflop computer also came on the scene in 1997.

The first conference to be called the *International* Computational Accelerator Physics Conference was held in Monterey, California in 1998, organized by me and Kwok Ko [29]. Parallel processing is highly evident in these proceedings. Also, Python begins to be seen, mentioned in a paper by Grote, Friedman, and Haber called "New Methods in WARP."

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

2000's: TERASCALE ERA

In the 2000's there were five ICAP conferences: Darmstadt in 2000, East Lansing in 2002, St. Petersburg in 2004, Chamonix in 2006, and San Francisco in 2009 [30–34]. As is evident, the conference venue had become truly international. During this decade there were major advances both in single-particle beam dynamics codes and in large-scale multi-physics beam dynamics codes. The DOE SciDAC program started in 2002. The IMPACT code suite was fully developed at the Berkeley Lab by J. Qiang [35]. The first version of Synergia was developed at Fermilab by Panagiotis Spentzouris and James Amundson [36]. It combined portions of IMPACT with the Leo Michelotti's mxyzptlk/beamline libraries. Andreas Adelmann developed the OPAL library [37]. Bmad was developed by David Sagan and others [38]. The Polymorphic Tracking Code, PTC, was developed by Etienne Forest [39]. The first million particle strong-strong beam-beam simulation was performed in 2004 [40]. The first billion particle linac simulation was performed in 2007 [41]. The ACE3P package was developed at SLAC, led by Kwok Ko, and was able to perform very high accuracy electromagnetic calculations involving extremely complicated 3D structures [42]. The first petaflop computer appeared in 2008.

2010's: PETASCALE ERA

Following ICAP 2009 it was decided to have the conference on a three-year cycle. So in the 2010's there have been three conferences, Rostock in 2012, Shanghai in 2015, and this meeting in Key West in 2018 [43–45].

During that time we've seen the meaning of "large scale" grow from tens of thousands of processes in the previous decade to hundreds of thousands in the current decade. And the very largest scale simulations now exceed a million processes.

Big Data emerged has as a major paradigm. Though the accelerator community's design needs don't usually involve it, our experiments, like those at places such as LHC, RHIC, and the light sources have helped drive developments in Big Data. It's well known that in fields like Cosmology, observations are quickly analyzed on supercomputers where the results drive the direction of observational resources. But that's happening in Accelerator Science too. For example, there is now a data pipeline between light source experiments and the NERSC supercomputer center.

Multi-level parallelization has grown increasingly dominant during this decade. This includes multiple levels of MPI, or MPI with threads, or MPI across nodes with hardware acceleration on a node. Multi-level MPI, in particular, has provided a relatively easy path to parallel parameter scans and parallel design optimization. Parallel design optimization has become one of the main uses of large-scale modeling, with many people using genetic optimizers.

PRESENT DAY, INTO THE FUTURE

I'll begin this final section of my talk by describing something that we are doing now, namely, performing 3D simulations of coherent synchrotron radiation.

As I look back I see the 2000's as a kind of Golden Era in space-charge modeling. Over the course of that decade, several multi-physics parallel beam dynamics code emerged that had 3D space-charge capability.

Now we are at the beginning of such an era in radiation modeling. This problem is extremely challenging. Consider that an N-body space-charge calculation requires N^2 operations; an analogous Lienard-Wiechert calculation would require N^2 operations but its difficulty is compounded by the fact that it would include the time-history of all particles. The physics of the problem further complicates the situation because the radiation cone is extremely narrow at high energy. In the past I have described this as being like a large number of flashlights that interact when their narrow light beams collide, taking into account light travel time.

One method for addressing this problem is known as the Lienard-Wiechert Particle-Mesh (LWPM) method [46]. This approach extends the widely used convolution-based method for modeling space-charge, but replaces the Coulomb Green function with the Lienard-Wiechert Green function.

It is well known that the most common method of computing 3D space charge in unbounded systems is to perform an FFT-based discrete convolution of a charge density with a Green function,

$$\phi_{i,j,k} = \frac{\delta_x \delta_y \delta_z}{4\pi\epsilon_o} \sum_{i'=1}^{i'_{max}} \sum_{j'=1}^{j'_{max}} \sum_{k'=1}^{k'_{max}} \rho_{i',j',k'} G_{i-i',j-j',k-k'}, \quad (1)$$

where $(\delta_x, \delta_y, \delta_z)$ is the grid cell size, $\rho_{i,j,k}$ is the charge density at the grid points, and $G_{i-i',j-j',k-k'}$ denotes G at values of grid point separation. Naively a convolution would scale as N^2, but because the approach is FFT-based it scales as $N \log N$.

In the case of a space-charge modeling code the quantity G is just the Coulomb Green function for the potential or the fields. The calculation of the G would then require only a few floating point operations (flops). The model can be made much more robust by using an Integrated Green function (IGF) instead of using the value of the "bare" Green function at the grid points [47]. Even so, the calculation of the IGF requires just a modest number of flops.

The transition to a model that includes both space-charge and radiation begins with the following observation: In space-charge codes the process is usually described as transforming the particles to the bunch frame where the motion is non-relativistic, solving for the potential or field on a grid, and transforming back to the lab frame. But the procedure can also be viewed as using the Heaviside representation of the Green function in the lab frame. In the case of the potential,

$$G_{\phi,heav} = \frac{1}{\gamma^2 r} \frac{1}{\left(1 - |\boldsymbol{\beta} \times \hat{\mathbf{r}}|^2\right)^{1/2}}, \quad (2)$$

13th Int. Computational Accelerator Physics Conf. ICAP2018, Key West, FL, USA JACoW Publishing
ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

where $\hat{\mathbf{r}}$ points from the (instantaneous) position of the charge to the observation point.

The transition to Lienard-Wiechert modeling replaces the Heaviside Green function – which is based on straight line motion at constant velocity – with the full Lienard-Wiechert Green function. For example, in the case of the electric field,

$$\mathbf{G}_{lw} = \left[\frac{q}{\gamma^2 \kappa^3 R^2} \left(\hat{n} - \vec{\beta} \right) + \frac{q}{\kappa^3 R c} \hat{n} \times \left\{ \left(\hat{n} - \vec{\beta} \right) \times \frac{\partial \vec{\beta}}{\partial t} \right\} \right]_{\text{ret}} \quad (3)$$

Here, $\hat{n} = \mathbf{R}/|R|$ is a unit vector pointing from the retarded emission point to the observation point, $\boldsymbol{\beta} = \mathbf{v}/c$, c is the speed of light, $\gamma = 1/\sqrt{1-\beta^2}$, and $\kappa = 1 - \hat{n} \cdot \boldsymbol{\beta}$.

But because it is preferable to use an IGF, one should embed this capability within a 3D quadrature package. So, to compute the Green function in a LWPM code, one calls a Lienard-Wiechert solver potentially millions of times to compute the Green function. This would be absolutely impossible in a serial code. But in a parallel code it is quite effective. It is even a good fit to current architectures because it involves a huge number of flops but basically no data movement to compute the Green function.

Previous studies have shown that the LWPM method agrees well with brute-force Lienard-Wiechert summation for the case of steady-state dipole radiation and for the case of a bunched beam inside a wiggler magnet [48]. Recently we have looked at the dipole example in a regime where the Coulomb field and the radiation field are comparable. Figure 3 shows the transverse electric field. This example corresponds to a 40 MeV electron bunch in a 0.16 T magnetic field. The bunch is Gaussian with an rms bunch size of 100 micron in x, y, and z. In this case the summation used 1 billion simulation particles. The summation and convolution results are in excellent agreement.

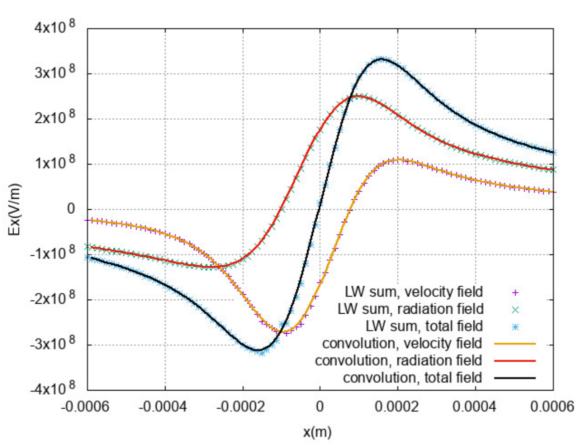

Figure 3: E_x *vs* x for the steady-state dipole test problem, plotted along the x-axis going through the bunch center. The LW velocity field, LW radiation field, and total field are shown. Results are shown for the LW summation over 1B particles and for the convolution-based method.

A 2D plot of the magnitude of the total transverse field is shown in Fig. 4. Note that magnitude is slightly larger for positive x, and there is a slight tilt in the dark band with respect to the line $x = 0$. These features would not be present in a space-charge code, i.e., a code based on the Heaviside approximation.

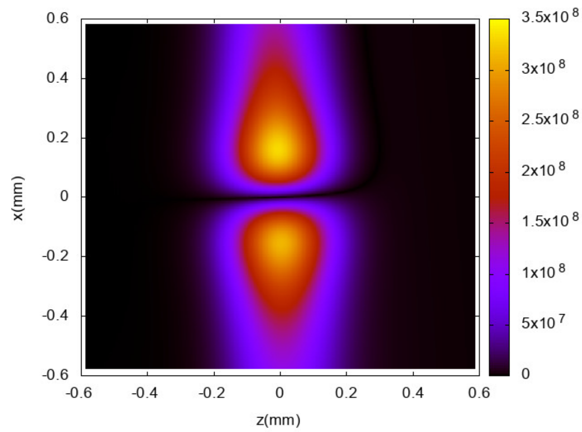

Figure 4: $|E_{x,total}|$ in the midplane $(x, 0, z)$ for the steady-state dipole test problem. The combined field does not show the signficant asymmetry that was present in $|E_{x,vel}|$ and $|E_{x,rad}|$ separately, but the field is still slightly larger in magnitude at positive x. Also, there is a slight tilt visible in the dark band with respect to the line $x = 0$.

To conclude this section I will discuss plans for exascale simulation in the 2020's. But first consider the following: At the time of the first Computational Accelerator Physics Conference in 1988, the fastest computer in the world was the Cray Y-MP. It had 8 vector processors running at 167 MHz and a performance of around 2 Gflops. Ten years later, at the time of ICAP'98 in Monterey, the teraflop barrier had recently been broken (in 1997) by the Intel ASCI Red computer built under the Accelerated Strategic Computing Initiative. Ten years after that, a year before ICAP'09 in San Francisco, the IBM Roadrunner computer at Los Alamos broke the petaflop barrier. And immediately people were thinking about the next big advance, as is clear from this ComputerWorld headline on June 9, 2008: "All hail Roadrunner's petaflop record; now, what about the exaflop?"

Now we're at ICAP 2018. Here in the USA there's a project called the Exascale Computing Project (ECP) [49]. It covers many scientific fields. One of the fields is advanced particle accelerator design, particularly plasma accelerator design [50]. Plasma accelerators have the potential to greatly reduce the size and cost of accelerators, with profound consequences for science and society. They may also provide a novel and economically viable path to the high-energy frontier through a plasma-based collider.

This is a case where large-scale modeling serves multiple purposes: First, it is a tool of discovery that allows us to explore the complex physical processes occurring in plasma accelerators. In some cases these processes may be extremely difficult or impossible to access experimentally, or it may be very expensive and time-consuming to do so. Exploration of plasma accelerators via large-scale

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

simulation will lead to insights that would otherwise be inaccessible. Second, it allows us to examine the feasibility of advanced concepts like plasma-based colliders. At present such simulations are too slow for rapid and thorough exploration of the parameter space. Under ECP, through advanced hardware and advanced algorithms able to run effectively on that hardware, such simulations will be possible in the early 2020's, which is the same time that exascale systems will become available. Lastly, large-scale simulation allows us to optimize the design of advanced accelerator concepts, and to develop designs that reduce cost and risk.

As we march toward the exascale era the computing environment is changing. Users requiring the most massive resources will soon have a fraction of an exaflop at their fingertips. Importantly, medium-scale users will also have increased computer power. To get a major boost in performance we will have to write code for heterogeneous hardware (CPUs, GPUs, etc.), using a mix of computer programming methods. In addition software libraries are already being written with a view toward exascale. For example, under ECP the Center for Particle Applications is developing a number of software libraries including libraries for parallel FFTs [51, 52]. Already FFTs can be performed on hundreds of thousands of cores for problem sizes up to $10,000^3$.

To conclude this section I will mention an example from another field, Computational Cosmololgy. Under the ExaSky ECP project [53], a simulation was performed on 1.5 million cores of the Sequoia computer at a sustained performance of nearly 14 petaflops, and used 3.6 trillion simulation particles. Such high resolution simulations are needed to make comparisons with high precision experimental measurements. The simulations are used to solve inverse problems to determine several key cosmological parameters like the amount of dark matter, parameters of primoridal fluctuations, etc.

CONCLUSION

I will conclude my talk by quoting something that I presented 10 years ago at the 2008 European Particle Accelerator Conference [54]. It is found in the Proceedings of the 1971 International Conference on High Energy Physics [55]. In response to a talk by Viktor Weisskopf, Lew Kowarski (who I mentioned previously) made a comment that was recorded in the Proceedings. Weisskopf had described the emergence of "a new type of physicist…the machine physicists,…" In the question and answer session Kowarski spoke. According to the proceedings he said,

"Early experimentalists worked with their hands: Galileo's legendary tossing of stones from the Tower of Pisa, or the alchemists mixing by hand the ingredients in their mixing bowls. In a similar way the theoreticians manipulated their numerical quantities and symbols by their unaided brain-power. Then came the machines to extend the experimenter's manual skill and to open whole new worlds of things to be handled in ways nobody could predict or even imagine before they really got going. Now we are at the beginning of a new kind of extension by machine: the computer comes to supplement the theoretician's brain. We cannot foresee what this fourth kind of creativity in physics will bring…"

This comment was made nearly 50 years ago when the fastest computer in the world was the CDC 7600 with a performance of about 10 Mflops. Sometime in the 2020's we will have exascale resources that have 100 *trillion* times the computing power that Kowarski knew 1971. Such a mind-boggling increase in computing power would have been almost unimaginable in 1971, and validates Kowarski's comment that "we cannot foresee what this fourth kind of creativity in physics will bring."

At conferences like this ICAP conference we share our experiences of what this fourth kind of creativity has brought to our field. More than ever, advanced computational modeling is enabling major advances and discoveries in Accelerator Physics. Opportunities abound in concepts like laser, plasma, and dielectric accelerators, in new approaches like integrable optics, in accelerator control and operation, in concepts for future colliders and future light sources, and in applications of accelerators.

ACKNOWLEDGEMENTS

The author thanks Salman Habib, John Shalf, and Jean-Luc Vay for helpful discussions. The simulation results presented here used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-05CH11231.

REFERENCES

[1] *Linear Accelerator and Beam Optics Codes*, Charles R. Eminhizer, Ed., AIP Conference Proceedings, vol. 177, 1988.

[2] F. T. Cole, "Oh, Camelot! A Memoir of the MURA Years", in *Proc. Cycl. Conf*, April 11, 1994. http://jacow.org/c01/cyc2001/extra/Cole.pdf

[3] T Dauxois , "Fermi, Pasta, Ulam, and a mysterious lady," *Physics Today*, vol. 61, pp. 55. https://arxiv.org/pdf/0801.1590.pdf

[4] H. Scott Dumas, '*The KAM Story: A Friendly Introduction to the Content, History, and Significance of Classical Kolmogorov-Arnold-Moser Theory*, World Scientific Publishing Company, 2014.

[5] *CERN Courier*, vol. 7, no. 9, September 1967.

[6] *CERN Courier*, vol. 12, no. 3, March 1972.

[7] K. L. Brown, "First and Second Order Matrix Theory for Design of Beam Transport Systems," Stanford Linear Accelerator Center, report SLAC-75, 1967.

[8] Alex J. Dragt *et al.*, "Lie Algebraic Treatment of Linear and Nonlinear Beam Dynamics," *Ann. Rev. Nucl. Part. Sci.*, vol. 38, pp. 455-496, 1988.

[9] A. J. Dragt and J. M. Finn, "Lie series and invariant functions for analytic symplectic maps," *J. Math. Phys.*, vol. 17, pp. 2215, 1976.

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-SUPLG01

[10] A. Dragt and E. Forest, "Computation of nonlinear behavior of Hamiltonian systems using Lie algebraic methods," *J. Math. Phys.*, vol. 24, pp. 2734, 1983.

[11] R. D. Ryne and A. J. Dragt, "Numerical Computation of Transfer Maps using Lie Algebraic Methods," in *Proc. PAC'87*, Washington, DC, USA.

[12] A. J. Dragt, *Lie Methods for Nonlinear Dynamics with Applications to Accelerator Physics*, https://www.physics.umd.edu/dsat/.

[13] https://laacg.lanl.gov/.

[14] F. Christoph Iselin, "The MAD Program," in *Proc HEACC83*, Batavia, IL, USA.

[15] R. D. Ruth, "A canonical integration technique," *IEEE Trans. Nucl. Sci.*, NS-30(4), August 1983.

[16] E. Forest and R. D. Ruth, "Fourth-order symplectic integration," *Physica D*, vol. 43, no. 1, pp. 105-117, 1989.

[17] H. Yoshida, "Construction of higher order symplectic integrators," *Phys. Lett. A*, vol. 150, no. 5-7, pp. 262-268, 1990.

[18] E. Forest, J. Bengtsson, and M. F. Reusch, "Application of the Yoshida-Ruth techniques to implicit integration and multi-map explicit integration," *Phys. Lett. A*, vol. 158, no. 3, pp. 99-101, 1991.

[19] M. Berz, "Differential Algebraic Description of Beam Dynamics to Very High Orders," *Particle Accelerators*, vol. 24, pp. 109-124, 1989.

[20] http://cosyinfinity.org/.

[21] E. Forest, M. Berz, and J. Irwin, "Normal form methods for complicated periodic systems: A complete solution using Differential algebra and Lie operators," *Particles Accelerators*, vol. 24, pp. 91, 1989.

[22] *Proceedings of the Conference on computer Codes and the Linear Accelerator Community*, Richard K. Cooper, Ed., LANL Report No. LA-11857-C, 1990, https://inis.iaea.org/collection/NCLCollectionStore/_Public/21/094/21094506.pdf

[23] *Proc. CAP'93*, Robert Ryne, Ed., AIP Conference Proceedings, vol. 297, 1993.

[24] R. D. Ryne *et al.*, "The US DOE Grand Challenge in Computational Accelerator Physics," in *Proc. Linac'98*, Chicago, USA, 1998.

[25] P. C. Liewer and V. K. Decyk, "A general concurrent algorithm for plasma particle-in-cell simulation codes," *J. Comput. Phys.*, vol. 85, pp. 302, 1989.

[26] R. D. Ryne, S. Habib, and T. P. Wangler, "Halos of Intense Proton Beams," in *Proc. PAC'95*, Dallas, TX, USA, 1995.

[27] R. D. Ryne and S. Habib, "Beam Dynamics Calculations and Particle Tracking using Massively Parallel Processors," in *Proc. LHC'95*, Montreux, Switzerland, 1995.

[28] *1996 Computational Accelerator Physics Conference*, J. Bisognano and A. Mondelli, Eds., AIP Conference Proceedings, vol. 391, 1997. http://inspirehep.net/record/458928

[29] *Proc. ICAP'98*,' K. Ko and R. D. Ryne, Eds., http://www.slac.stanford.edu/econf/C980914/papers/ICAP98_eConf.pdf

[30] http://prst-ab.aps.org/speced/ICAP2000/.

[31] http://www.bt.pa.msu.edu/ICAP02/main.html

[32] *Proc ICAP'04,*, St. Petersburg, Russia, June 29-July 2, 2004 D. Ovsyannikov, M. Berz, K. Makino, Eds., *Nucl. Instrum. Meth.* A558, 2006, pp.1-365.

[33] http://jacow.org/ICAP06/.

[34] http://jacow.org/ICAP2009/.

[35] J. Qiang *et al.*, "An object-oriented parallel particle-in-cell code for beam dynamics simulation in linear accelerators," *J. Comput. Phys.*, vol. 163, no. 2, pp. 434-451, 2000.

[36] J. Amundson, P. Spentzouris, J. Qiang, and R. Ryne, "Synergia: An accelerator modeling tool with 3-D space charge," *J. Comput. Phys.*, vol. 211, no. 1, pp. 229-248, 2006.

[37] A. Adelmann *et al.*, "The object-oriented parallel accelerator library (OPAL), design, implementation, and application," in *Proc. ICAP'09*, http://jacow.org/ICAP2009/papers/we3iopk01.pdf

[38] J. Urban, L. Fields, D. Sagan, "Linear Accelerator Simulations with Bmad," in *Proc. PAC'05*, Knoxville, TN, USA.

[39] E. Forest, Y. Nogiwa, and F. Schmidt, "The FPP Documentation", in *Proc. ICAP'06*, Chamonix, Switzerland.

[40] J. Qiang, M. A. Furman, and R. D. Ryne, "Strong-strong beam-beam simulation using a Green function approach," *Phys. Rev. ST Accel. Beams*, vol. 5, pp. 104402, 2002.

[41] J. Qiang, I. Pogorelov, and R. D. Ryne, "Parallel Beam Dynamics Simulation Tools for Future Light Source Linac Modeling," in *Proc. PAC'07*, Albuquerque, NM, USA.

[42] https://portal.slac.stanford.edu/sites/ard_public/acd/Pages/Default.aspx

[43] http://jacow.org/ICAP2012/.

[44] http://jacow.org/ICAP2015/.

[45] http://jacow.org/ICAP2018/.

[46] R. D. Ryne, J. Qiang, C. E. Mitchell, and B. E. Carlsten, "Self-Consistent Modeling using a Lienard-Wiechert Particle-Mesh Method," in *Proc. IPAC'18*, Vancouver, BC, Canada.

[47] J. Qiang, S. Lidia, R. D. Ryne, and C. Limborg-Deprey, "3D quasistatic model for high brightness beam dynamics simulation," *Phys. Rev. ST Accel. Beams*, vol. 9, pp. 044204, 2006.

[48] R. D. Ryne *et al.*, "Using a Lienard-Wiechert Solver to Study Coherent Synchrotron Radiation Effects," in *Proc. FEL'13*, New York, NY, USA.

[49] http://www.exascaleproject.org/.

[50] http://www.exascaleproject.org/project/warpx-exascale-modeling-of-advanced-particle-accelerators/.

[51] http://www.exascaleproject.org/project/copa-co-design-center-particle-applications/.

[52] S. Plimpton, "fftMPI, a distributed-memory parallel FFT library," https://fftmpi.sandia.gov/.

[53] http://www.exascaleproject.org/project/exasky-computing-sky-extreme-scales/.

[54] R. D. Ryne, "Advanced Computing Tools and Models for Accelerator Physics," in *Proc. EPAC'08*, Genoa, Italy.

[55] V. F. Weisskopf, " The Past and the Future of High Energy Physics", in *Proc. 8th International Conference on High-Energy Accelerators*, M. Hildred Blewett, ed., 1971.