Employing GPUs for numerical integration with Lie-series

András Pál

Konkoly Observatory of the Hungarian Academy of Sciences
Department of Astronomy, Loránd Eötvös University

E-mail address: apal@szofi.net

Abstract

The method of Lie-integration is a very effective algorithm for numerical solution of ordinary differential equations. The principle of the algorithm is to compute the coefficients of the Taylor-series for the solution involving recurrence relations. This approach also yields more possibilities for various types of adaptive integration since not only the integration stepsize but simultaneously the polynomial order of the power series expansion can also be altered. In addition, alternation of the stepsize does not yield a loss in the (expensive) computing time. The "disadvantage" of the method is because of the recurrence formulae: these set of equations depends on the particular problem itself and therefore had to be derived in advance of the actual implementation. However, the method is definitely faster than the classic known explicit methods (that do not depend on the right-hand side of the differential equation), has better error propagation properties and the "side-effect" of knowing the analytic expansion of the solution also allows us other kind of studies.

The previously mentioned recurrence relations are known for the N-body problem, thus the dynamical analysis of planetary systems could be made very effective. In this presentation we discuss the questions and possibilities related to the implementation of the Lie-integration algorithm on GPU architectures. We briefly summarize other advantages of this numerical method that makes particularly suitable on GPU systems. For instance, how the fact that the computation of the recurrence relations (in the case of the N-body problem) needs only evaluating additions, subtractions and multiplications can be exploited on GPUs. Initial works show that studies related to exploration of the phase space (thus as stability studies, where the similar dynamical system is investigated in the case of various initial conditions) can be achieved rather efficiently. Such studies are in the focus of astronomical research in the case of both the Solar System and extrasolar planetary systems as well.

The Lie-integration

- The study of various astrophysical phenomena and problems requires the solution of ordinary differential equations. Some of these are related to the (gravitational) N-body problem, thus having an efficient method for computing the numerical solution of the underlying equations can significantly aid the analysis of these astrophysical systems.
- In general, let us write the differential equation of our interest in the form of

$$\dot{x}_i = f_i(\mathbf{x}),$$

where $\mathbf{x} \equiv (x_1, \dots)$ is an $\mathbb{R} \to \mathbb{R}^N$ and $\mathbf{f} \equiv (f_1, \dots)$ is an $\mathbb{R}^N \to \mathbb{R}^N$ smooth function.

- Let us also introduce the operators

$$D_i := \frac{\partial}{\partial x_i} \qquad \text{and} \qquad L := \sum_{i=1}^{N} f_i \frac{\partial}{\partial x_i} = \sum_{i=1}^{N} f_i D_i$$

where the latter is known as the Lie-operator. L is a linear and a differential operator, i.e. $L(\lambda a + b) = \lambda L(a) + L(b)$ $L(ab) = aL(b) + bL(a)$ for all a and b variables and λ constants.

- It can easily be proven (see [1] or [2]) that the solution of the original equation, $\dot{x}_i = f_i(\mathbf{x})$ at a given instance $t + \Delta t$ is formally

$$\mathbf{x}(t + \Delta t) = \exp\left(\Delta t \cdot L\right) \mathbf{x}(t),$$

where

$$\exp\left(\Delta t \cdot L\right) = \sum_{k=0}^{\infty} \frac{\Delta t^k}{k!} L^k = \sum_{k=0}^{\infty} \frac{\Delta t^k}{k!} \left(\sum_{i=1}^{N} f_i D_i \right)^k.$$

- Lie-integration: it is the finite approximation of the above, up to the order M:

$$\mathbf{x}(t + \Delta t) \approx \left(\sum_{k=0}^{M} \frac{\Delta t^k}{k!} L^k \right) \mathbf{x}(t) = \sum_{k=0}^{M} \frac{\Delta t^k}{k!} \left(L^k \mathbf{x}(t) \right).$$

- Although the above seems to be trivial, the actual computation of the terms $L^k\mathbf{x}$ is not an easy task. In practice, the computation of these derivatives are based on recurrence relations: the terms $L^{n+1}\mathbf{x}$ are expressed as functions of $L^k\mathbf{x}$ where $0 \le k \le n$. The cardinal problem of Lie-integration is the derivation of these recurrence relations (thus, no explicit schemes are available like, e.g., for the Runge-Kutta method, see [3]).

The N-body problem

- Let us have N bodies of which motion are determined by the Newtonian gravitational law. Let us denote the mth component of the coordinates and velocities of the ith body by x_{im} and v_{im}, respectively (obviously, $1 \le i \le N$ and $m = 1$, 2 or 3).
- Using the calculations of [2] and [4], the recurrence relations can be written as

$$\begin{aligned}
L^{n+1} x_{im} &= L^n v_{im}, \\
L^n A_{ijm} &= L^n x_{im} - L^n x_{jm}, \\
L^n B_{ijm} &= L^n v_{im} - L^n v_{jm}, \\
L^{n+1} v_{im} &= -G \sum_{j \ne i} m_j \left[\sum_{k=0}^{n} \binom{n}{k} L^k \phi_{ij} L^{n-k} A_{ijm} \right], \\
L^n \Lambda_{ij} &= \sum_{k=0}^{n} \binom{n}{k} L^k A_{ijm} L^{n-k} B_{ijm}, \\
L^{n+1} \phi_{ij} &= \rho_{ij}^{-2} \sum_{k=0}^{n} F_{nk} L^{n-k} \phi_{ij} L^k \Lambda_{ij},
\end{aligned}$$

where $F_{nk} = (-3)\binom{n}{k} + (-2)\binom{n}{k+1}$ and we introduced ρ_{ij}, which is the distance between the ith and jth bodies and $\phi_{ij} = \rho_{ij}^{-3}$.

- In order to bootstrap these recurrence relations, we only have to use the fact that $L^0 a = a$, where a can be any of the quantities appearing above.
- Note: these recurrence relations can also be written in relative coordinates, i.e. when one of the bodies is fixed in the origin (that is useful for analysis of planetary systems or systems where one of the bodies has significantly larger mass than the other ones), see also [4] for further details.
- It can easily be seen that with the exception of the computation of $L^0\phi_{ij}$ and ρ_{ij}, the recurrence formulae have only addition, subtraction and multiplication operations. In addition, once the Lie-derivatives $L^k x_{im}$ and $L^k v_{im}$ are known up to an order of $k \le M$, the computation of the numerical approximation of the Lie-exponent requires also purely addition and multiplication.
- Therefore, the complete integration method based on the Lie-series can effectively be implemented on architectures which are optimized for these elementary arithmetic operations (addition, subtraction and multiplication). Since graphics processing units (GPUs) are designed to compute these elementary operations in parallel, in the following we investigate the possibilities and properties related to the implementation of Lie-integration on such architectures.

Computation needs and GPUs

- We can easily estimate the computing requirements of the Lie-integration of gravitational N-body systems. Basically, the computation of these terms are done in two steps:
  - The first step is the bootstrap procedure when we initialize the variables $L^0 a$. Of course, the number of required operations in the bootstrap computations does not depend on the integration order. The computation of the terms ρ_{ij} require $(3/2)N(N-1)$ subtractions and $N(N-1)/2$ multiplications and the same number of square root operations while the computation of ρ_{ij}^{-2} and $\phi_{ij} = \rho_{ij}^{-3}$ needs $N(N-1)$ divisions additionally. All in all, these computation costs scale as $\mathcal{O}(N^2)$ for large number of bodies.
  - The second step is the evaluation of the recurrence relations. In a similar manner, one can obtain that (due to the presence of the summations in the expressions for v_{im} and Λ_{ij}) the computation time scales as $\mathcal{O}(N^2M^2)$. The most time-consuming part is the inner sum in the former one while the time needed by all of the other terms scales as $\mathcal{O}(N^2M)$, $\mathcal{O}(NM^2)$ or simply $\mathcal{O}(NM)$.
- In the scientific practice of planetary dynamics, the number of bodies can be so small that we are not interested in the asymptotic dependence of the computing time as the function of N. However, as it is shown by [4] or [5], the optimal integration order can be as large as $M \approx 20 \dots 25$, depending on the actual problem and the desired precision. Therefore optimization is essential only for the terms of which total computation time scales as $\mathcal{O}(M^2)$. In other words, the number of complex operations (square root or division) is always negligible compared to the number of multiplications and additions.
- In addition, the recurrence relations does not need the actual value of ρ_{ij}, in the set of the recurrence relations, only the terms ρ_{ij}^{-2} and ρ_{ij}^{-3} appear. Therefore, if the time evolution of these ρ_{ij}^p quantities are also computed using the respective ordinary differential equations (and the respective Lie-series), one can also eliminate the need of computing square roots or performing divisions during the bootstrap procedure. Similarly to the proof presented in [4], it can be shown that

$$L^{n+1} \rho_{ij}^p = \rho_{ij}^{-2} \sum_{k=0}^{n} G_{nk}^{(p)} L^{n-k} \rho_{ij}^p L^k \Lambda_{ij},$$

where $G_{nk}^{(p)} = p\binom{n}{k} + (-2)\binom{n}{k+1}$. Thus, we can conclude that the *whole* numerical integration can be implemented with Lie-series by fully eliminating the square root and division operations. However, this elimination has a computing cost that scales also as $\mathcal{O}(NM^2)$, but does not significantly increase the total computing time.

Conclusions

- The numerical integration based on Lie-series is a very efficient and elegant way of analyzing the gravitational N-body problem. Long-term stability investigations and chaos detection of the Solar System and other planetary systems require to solve the similar set of equations for thousands or millions of initial conditions.
- A full implementation of the Lie-integration of gravitational N-body problem is available from the address `http://http://szofi.elte.hu/~apal/utils/astro/lieint/`. The *core* of this code can be ported to Nvidia's CUDA (GP)GPU architecture without any changes.
- If the number of bodies, N is fixed and relatively small, all of the variables and arrays can be stored efficiently in the multiprocessor registers.
- Analysis of independent initial conditions of the same problem (set of equations) can be done in parallel without any practical need for thread synchronization.
- Although as of now, the full implementation on GPUs are in an initial stage, this method seems to be a promising alternative for dynamical analysis.

References

[1] Gröbner, W. & Knapp, H.: Contributions to the Method of Lie-Series, 1967, Bibliographisches Institut, Mannheim

[2] Hanslmeier, A. & Dvorak, R., 1984, A&A, 132, 203

[3] Press, W. H., Teukolsky, S. A., Vetterling, W.T., Flannery, B.P., 1992, Numerical Recipes in C: the art of scientific computing, Second Edition, Cambridge University Press

[4] Pál, A. & Süli, Á., 2007, MNRAS, 381, 1515

[5] Pál, A., 2010, MNRAS, 409, 975

Acknowledgements

The work of the author have been supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and by the ESA grant PECS 98073.

LaTeX typeset