ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-TUP097

THE IMPLEMENTATION OF ADAPTIVE STEP SIZE RUNGE KUTTA INTEGRATOR IN ZGOUBI

Jonathan Lee, Stony Brook University, Stony Brook, NY, USA
Bhawin Dhital, Haixin Huang, François Méot, Brookhaven National Laboratory, Upton, NY, USA

Abstract

The Zgoubi simulation code for beam and spin dynamics employs a numerical method based on Taylor series to integrate the Lorentz and Thomas-BMT equations, optimizing computational efficiency while ensuring high accuracy and robust preservation of motion invariants. In this work, we developed and implemented an adaptive step-size Runge-Kutta (RK) integrator for implementation into Zgoubi to tackle growing computational demands in accelerator physics simulations. This new integrator complements Zgoubi's default solver, offering users the flexibility to choose between integration methods based on specific simulation requirements. We demonstrated that the adaptive step-size RK integrator achieves the necessary accuracy and performance for integrating the Lorentz and Thomas-BMT equations effectively. A key advantage of Zgoubi lies in its wide optical elements library, featuring over 60 accelerator components and variants, which the new adaptive step-size RK integrator can seamlessly utilize. Developed and rigorously tested over decades across numerous projects, this library provides a high degree of confidence in the code's reliability. The same advantage holds about ancillary computations such as synchrotron radiation, space charge, decay in flight, etc. The implementation of the adaptive step-size RK integrator supports Zgoubi's adaptability, enabling simulations of complex beam and spin dynamics with a trusted and well-established computational framework.

INTRODUCTION

In numerical analysis, solving ordinary differential equations (ODEs) is a critical task that requires methods balancing accuracy and computational efficiency. ODEs typically take the form:

$$\frac{dy}{dt} = f(t, y), \quad y(t_0) = y_0, \tag{1}$$

where $f(t, y)$ defines the dynamics, and $y(t_0) = y_0$ is the initial condition. Among the most prominent techniques for approximating solutions are the Runge-Kutta methods, a family of iterative algorithms. Adaptive step size methods address the inefficiencies of fixed-step integrators by dynamically adjusting the integration step size h based on the local behavior of the solution. These methods estimate the error at each step and modify the step size to keep the error within a specified tolerance. A common approach is to use embedded Runge-Kutta pairs, where two methods of different orders (e.g., 4th and 5th) share function evaluations. The difference between their solutions provides an estimate of the local truncation error, enabling step size adjustment.

The storage ring proton Electric Dipole Moment (srpEDM) experiment leverages a novel symmetric-hybrid, frozen-spin storage ring design using radial electric bending and magnetic focusing, enabling precise measurement of the vertical rotation of the polarization of a highly-polarized stored proton beam, targeting a sensitivity of 10^{-29} e·cm [1]. A primary challenge lies in achieving high integration accuracy in electric (or combined electric and magnetic) fields, where varying rigidity complicates simulations. To address this challenge and the increasing computational demands in accelerator physics, we developed and implemented an adaptive step-size Runge-Kutta integrator (Tsit5 method) within the Zgoubi framework [2, 3], enhancing the precision of simulations in electric and combined E-B fields.

TSIT 5TH ORDER ADAPTIVE STEP SIZE RUNGE KUTTA SOLVER

The Tsit5 method [4], developed by Tsitouras, is a sophisticated 5th-order Runge-Kutta integrator with an embedded 4th-order method, often denoted as a 5(4) pair. This design allows Tsit5 to compute a 5th-order accurate solution while simultaneously producing a 4th-order solution for error estimation, all with shared function evaluations for efficiency. Unlike RK4, Tsit5 adapts the step size h at each iteration, reducing it in regions of high curvature and increasing it where the solution is smooth. This adaptability enhances both accuracy and computational efficiency, making Tsit5 particularly well-suited for ODEs with varying dynamics. Its construction, detailed in a Butcher tableau, specifies the coefficients for its stages, but the key advantage lies in its ability to maintain high-order accuracy while optimizing performance across diverse applications.

The Tsit5 method advances the solution from t_n to $t_{n+1} = t_n + h$ using the following scheme:

$$y_{n+1} = y_n + h \sum_{i=1}^{6} b_i k_i, \tag{2}$$

and an embedded 4th-order estimate:

$$\hat{y}_{n+1} = y_n + h \sum_{i=1}^{6} \hat{b}_i k_i, \tag{3}$$

where the stages k_i are computed as:

$$\begin{aligned}
k_1 &= f(t_n, y_n), \\
k_2 &= f(t_n + c_2 h, y_n + h a_{21} k_1), \\
k_3 &= f(t_n + c_3 h, y_n + h(a_{31} k_1 + a_{32} k_2)), \\
k_4 &= f(t_n + c_4 h, y_n + h(a_{41} k_1 + a_{42} k_2 + a_{43} k_3)), \\
k_5 &= f(t_n + c_5 h, y_n + h(a_{51} k_1 + a_{52} k_2 + a_{53} k_3 + a_{54} k_4)), \\
k_6 &= f(t_n + c_6 h, y_n + h(a_{61} k_1 + a_{62} k_2 + a_{63} k_3 + a_{64} k_4 + a_{65} k_5)).
\end{aligned}$$

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-TUP097

The coefficients c_i, a_{ij}, b_i, and $\hat{b}_i$ are chosen to achieve the desired order of accuracy. The coefficients for the Tsit5 method are given in the following Butcher tableau form [5]:

$$\begin{array}{c|ccccc}
0 & & & \cdots & \\
c_2 & a_{21} & & & \\
c_3 & a_{31} & a_{32} & & \\
\vdots & \vdots & \vdots & \ddots & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} \\
\hline
 & b_1 & b_2 & \cdots & b_s \\
 & \hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_s
\end{array}$$

WHAT TO BE SOLVED? LORENTZ EQUATIONS AND THOMAS-BMT EQUATIONS

The particle 6D coordinates (x, a, y, b, l, δ) as [6]

$$a = \frac{p_x}{p_0} \tag{4a}$$
$$b = \frac{p_y}{p_0} \tag{4b}$$
$$l = \kappa(t - t_0) \tag{4c}$$
$$\delta = \frac{K - K_0}{K_0}, \tag{4d}$$

where x is the horizontal position, a is the horizontal momentum slope, y is the vertical position, b is the vertical momentum slope, l is a time-of-flight like variable, with $\kappa = -v_0\gamma_0/(1+\gamma_0)$, v_0 is the velocity of the particle, γ_0 is the Lorentz factor. δ is the energy deviation.

Given the variables $x = u_1$, $a = u_2$, $y = u_3$, $b = u_4$, $l = u_5$, $\delta = u_6$, $S_x = u_7$, $S_y = u_8$, $S_s = u_9$, the following equations are defined [6]:

$$\frac{du_1}{ds} = \frac{a(1+hx)}{\frac{p_s}{p_0}}, \tag{5a}$$
$$\frac{du_2}{ds} = (1+hx)\left[\frac{1+\eta}{1+\eta_0}\frac{E_x}{\chi_{E0}\frac{p_s}{p_0}} + \frac{bB_s}{\chi_{M0}\frac{p_s}{p_0}} - \frac{B_y}{\chi_{M0}}\right] + h\frac{p_s}{p_0}, \tag{5b}$$
$$\frac{du_3}{ds} = \frac{b(1+hx)}{\frac{p_s}{p_0}}, \tag{5c}$$
$$\frac{du_4}{ds} = (1+hx)\left[\frac{1+\eta}{1+\eta_0}\frac{E_y}{\chi_{E0}\frac{p_s}{p_0}} + \frac{B_x}{\chi_{M0}} - \frac{aB_s}{\chi_{M0}\frac{p_s}{p_0}}\right], \tag{5d}$$
$$\frac{du_5}{ds} = \frac{(1+hx)m_p(1+\eta)}{p_s}, \tag{5e}$$
$$\frac{du_6}{ds} = 0, \tag{5f}$$
$$\frac{du_7}{ds} = \Omega_y u_9 - \Omega_s u_8, \tag{5g}$$
$$\frac{du_8}{ds} = \Omega_s u_7 - \Omega_x u_9, \tag{5h}$$
$$\frac{du_9}{ds} = \Omega_x u_8 - \Omega_y u_7. \tag{5i}$$

with

$$\frac{p_s}{p_0} = \sqrt{\frac{\eta(2+\eta)}{\eta_0(2+\eta_0)} - a^2 - b^2} \tag{6a}$$
$$\eta = \eta_0(1+\delta) - \frac{ZeV}{m_0c^2} \tag{6b}$$
$$\eta_0 = \gamma_0 - 1 \tag{6c}$$
$$\chi_{m0} = \frac{p_0}{Ze} \tag{6d}$$
$$\chi_{e0} = \frac{p_0 v_0}{Ze} \tag{6e}$$
$$h(s) = \frac{d\theta(s)}{ds} \tag{6f}$$
$$K_0 = \eta_0 m_0 c^2, \tag{6g}$$

where χ_{m0} is the magnetic rigidity, and χ_{e0} is the electric rigidity, $h(s)$ is the curvature, K_0 is the kinetic energy of the particle, p_0 is the momentum of the particle, E_x, E_y, E_s is the horizontal, vertical and longitudinal electric field, and B_x, B_y, B_s is the horizontal, vertical and longitudinal magnetic field.

With

$$\Omega_x = -\frac{Ze}{m_0}\frac{dt}{ds}\Big\{\left[C_1B_x - C_2\beta_x\beta B - C_3(\beta_yE_s - \beta_sE_y)\right] + \frac{d_p}{2}\left[(\beta_yB_s - \beta_sB_y) + \frac{E_x}{c} - C_4\beta_x\beta E\right]\Big\}, \tag{7a}$$
$$\Omega_y = -\frac{Ze}{m_0}\frac{dt}{ds}\Big\{\left[C_1B_y - C_2\beta_y\beta B - C_3(\beta_sE_x - \beta_xE_s)\right] + \frac{d_p}{2}\left[(\beta_sB_x - \beta_xB_s) + \frac{E_y}{c} - C_4\beta_y\beta E\right]\Big\} + h, \tag{7b}$$
$$\Omega_s = -\frac{Ze}{m_0}\frac{dt}{ds}\Big\{\left[C_1B_s - C_2\beta_s\beta B - C_3(\beta_xE_y - \beta_yE_x)\right] + \frac{d_p}{2}\left[(\beta_xB_y - \beta_yB_x) + \frac{E_s}{c} - C_4\beta_s\beta E\right]\Big\} \tag{7c}$$

and

$$\frac{dt}{ds} = \frac{(1+hx)m_0(1+\eta)}{p_s}, \tag{8a}$$
$$p_x = ap_0,\ p_y = bp_0, \tag{8b}$$
$$\beta_x = \frac{p_x}{\gamma_0 m_0 c},\ \beta_y = \frac{p_y}{\gamma_0 m_0 c},\ \beta_s = \frac{p_s}{\gamma_0 m_0 c}, \tag{8c}$$
$$C_1 = G_p + \frac{1}{\gamma_0},\ C_2 = \frac{G_p\gamma_0}{\gamma_0 + 1}, \tag{8d}$$
$$C_3 = \frac{G_p + \frac{1}{\gamma_0+1}}{c},\ C_4 = \frac{\gamma_0}{c(\gamma_0+1)}, \tag{8e}$$
$$\beta B = \beta_xB_x + \beta_yB_y + \beta_sB_s, \tag{8f}$$
$$\beta E = \beta_xE_x + \beta_yE_y + \beta_sE_s, \tag{8g}$$

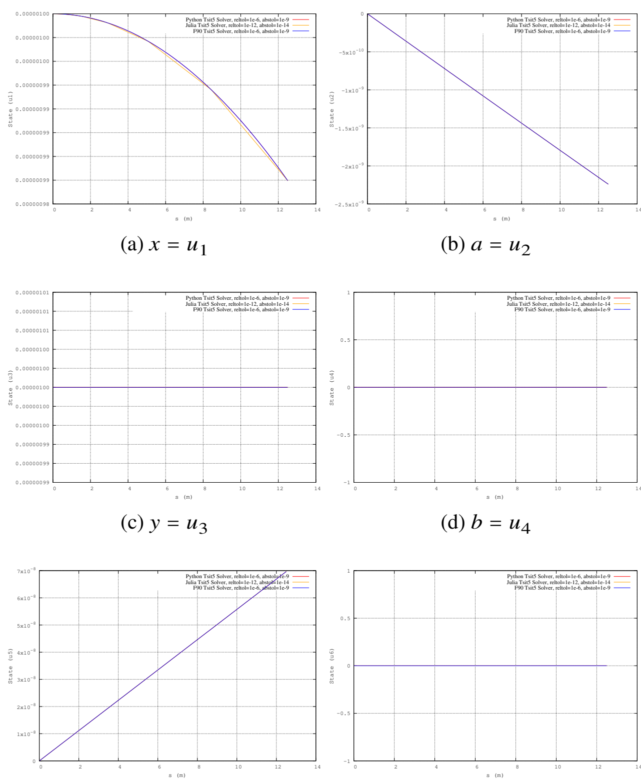
where d_p is the Electric Dipole Moment (EDM) of particle, G_p is the anomalous magnetic dipole moment of particle.

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-TUP097

IMPLEMENTATIONS AND VALIDATIONS

The Tsit5 solver was initially implemented in Python for prototype verification and subsequently rewritten in Fortran 90 to allow for integration within the existing Zgoubi Fortran codebase. The SpinTrack simulation code [7], developed by Z. Omarov, employs the Tsit5 solver from Julia's DifferentialEquations.jl package to integrate the Lorentz and Thomas-BMT equations.

This study integrates the Lorentz and Thomas-BMT equations to model beam and spin dynamics within an electrostatic deflector, assuming no fringe fields, as a validation case [3]. Numerical integration results and comparisons for this example, obtained using Python, Fortran 90, and the Julia Tsit5 solver, are presented in Fig. 1 (Lorentz equations) and Fig. 2 (Thomas-BMT equations). Detailed numerical results and comparisons for the final integration step, computed using the Python, F90, and Julia Tsit5 solvers, are provided in Table 1. The particle is initially launched from $x = 1\,\mu\mathrm{m}$, $a = 0\,\mathrm{rad}$, $y = 1\,\mu\mathrm{m}$, $b = 0\,\mathrm{rad}$, $l = 0\,\mathrm{m}$, $\delta = 0$, $S_x = 0$, $S_y = 0$, $S_z = 1$.



(a) $x = u_1$ (b) $a = u_2$

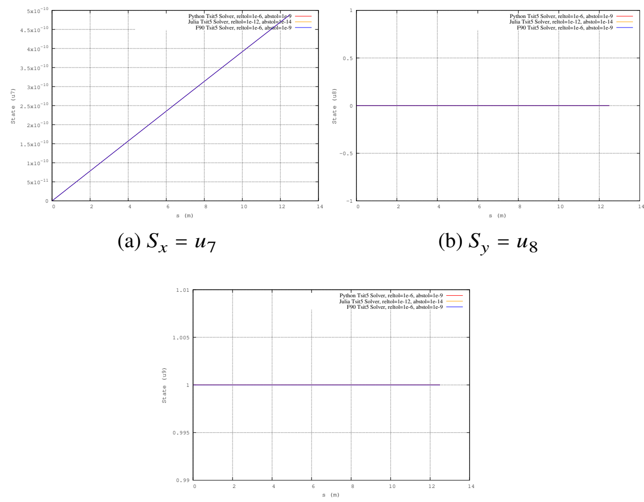
(c) $y = u_3$ (d) $b = u_4$

(e) $l = u_5$ (f) $\delta = u_6$

Figure 1: Comparison of Tsit5 Solver Results for Lorentz Equations within an Electrostatic Deflector: Python, Fortran 90, and Julia Implementations

CONCLUSION

All three Tsit5 solvers implemented in this study demonstrate exceptional precision and accuracy in integrating the Lorentz and Thomas-BMT equations. Efforts to integrate and test this solver within the Zgoubi codebase are currently underway.



(a) $S_x = u_7$ (b) $S_y = u_8$

(c) $S_s = u_9$

Figure 2: Comparison of Tsit5 Solver Results for Thomas-BMT equations within an Electrostatic Deflector: Python, Fortran 90, and Julia Implementations.

Table 1: Final Step Results Comparison of Tsit5 Solver for Lorentz and Thomas-BMT Equations within an Electrostatic Deflector: Python, Fortran 90, and Julia Implementations.

Solver	**Final Step u1, u2, u3, u4, u5, u6, u7, u8, u9 Value (s=12.5)**
Python Tsit5 Solver, reltol=1e-6, abstol=1e-9	9.859649595698982e-07 -2.240338755615162e-09 9.999999999999999e-07 0.000000000000000e+00 6.968076175342249e-08 0.000000000000000e+00 4.885494704934485e-10 0.000000000000000e+00 1.000000000000000e+00
F90 Tsit5 Solver, reltol=1e-6, abstol=1e-9	9.859649595100790E-07 -2.240338760270500E-09 1.000000000000000E-06 0.000000000000000E+00 6.968076175342250E-08 0.000000000000000E+00 4.885494355880850E-10 0.000000000000000E+00 1.000000000000000E+00
Julia Tsit5 Solver, reltol=1e-12, abstol=1e-14	9.859649591868269e-07 -2.240338846141274e-09 1.000000000000000e-06 0.000000000000000e-00 6.968076175342249e-08 0.000000000000000e-00 4.885494660604705e-10 0.000000000000000e-00 1.000000000000000e+00

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-TUP097

REFERENCES

[1] Z. Omarov *et al.*, "Comprehensive symmetric-hybrid ring design for a proton EDM experiment at below 10^{-29} $e\cdot$ cm", *Phys. Rev. D*, vol. 105, p. 032001, Feb. 2022. doi:10.1103/PhysRevD.105.032001

[2] F. Méot, "Zgoubi User's guide", Brookhaven National Laboratory (BNL), Upton, NY, USA, Rep. BNL-99611-2013-IR, Nov. 2012. doi:10.2172/1062013

[3] F. Méot, *Understanding the Physics of Particle Accelerators: A Guide to Beam Dynamics Simulations Using ZGOUBI*, Springer International Publishing, 2024. doi:10.1007/978-3-031-59979-8

[4] Ch. Tsitouras, "Runge–Kutta pairs of order 5(4) satisfying only the first column simplifying assumption", *Comput. Math. Appl.*, vol. 62, no. 2, pp. 770-775, 2011. doi:10.1016/j.camwa.2011.06.002

[5] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, Wiley, 2003. doi:10.1002/0470868279

[6] M. Berz, K. Makino, and W. Wan, *An Introduction to Beam Physics*, Boca Raton, FL, USA: CRC Press, 2014. doi:10.1201/b12074

[7] Z. Omarov, "SpinTrack.jl", GitHub, ver. 0.0.1, Aug. 2022, https://github.com/isentropic/SpinTrack.jl