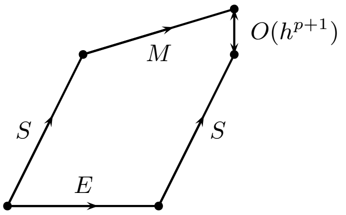
Blowing our own Google; Dynamical Systems and Numerical Analysis

I thought it meant something that, not so long ago, a Google search for John Butcher gave me first billing, and many of the remaining first few places as well. Now I have been displaced by John Butcher, an English Jazz Saxophonist. But I am still ahead of John Butcher, an American baseball player.

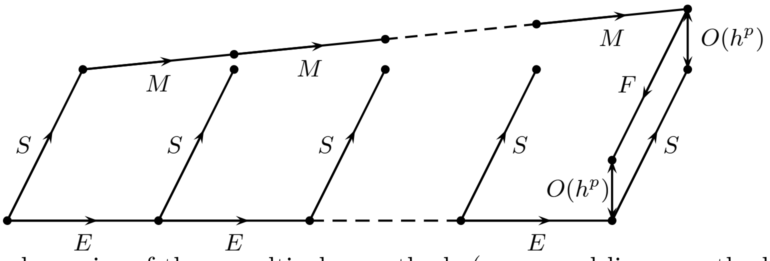
Another search, for Dynamical Systems and Numerical Analysis, reveals 463,000 hits and the first of these is for the NZIMA Thematic Programme with this title. This is evidently a gung ho subject and we in New Zealand must be a gung ho part of it. With all this resonance between Dynamical Systems and Numerical Analysis there must be something I can say about one or the other or possibly both. The best I can do is to make some comparisons between two distinct types of numerical methods for evolutionary problems and see if there is anything related to dynamical systems coming out of the comparison.

The two distinct approaches to numerical ordinary differential equations that I am referring to are known as one-step methods and multistep methods. Given an autonomous differential equation system characterised by the vector field f on a vector space X, the flow through a time interval h is often written as $\exp(hf) : X \to X$ but I will write it as E (with the dependence on h suppressed from the notation). A one-step method to approximate the action of E typically involves a number of evaluations of f and a recombination of the results. The standard method of this type is a Runge-Kutta method and I will write a typical example by the symbol R. Since R is supposed to approximate E, an important concept is the local truncation error which measures how much the result computed by R differs from the exact solution represented by the action of E. The order of the method is an integer p such that the local truncation error can be estimated in terms of h^{p+1}. Over an extended numerical integration, these local errors combine and reinforce each other leading to a global error bound in terms of h^p.

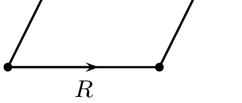
In parallel with the historical development of Runge-Kutta methods, multistep methods were achieving popularity for practical computations because of their low costs. However, on the face of it, they have much more complicated dynamics because the computations take place in the vector space X^r, where the integer r indicates how many items of information are passed from step to step. Introduce two mappings $S : X \to X^r$ and $F : X^r \to X$, where it is supposed that $F \circ S = \text{id}$. In a computer implementation of an r-value method, S (the "starting method") is used to generate input to the first step and F (the "finishing method") is used to produce a usable approximation to the solution. To understand the meaning of accuracy for this sort of method, it is not enough to simply estimate the quality of $F \circ M \circ S$ as an approximation to E, where $M : X^r \to X^r$ denotes a single step of the multivalue integration process, because we want to carry on for many steps applying F only at the end. Hence, we need to assess the accuracy in terms of $M \circ S$ as an approximation to $S \circ E$. This is shown in the schema at the right.



In a long-term computation, over n steps, armed with stability conditions imposed on any convergent method, it is possible to estimate the error in $M^n \circ S$, as an approximation to $S \circ E^n$. Now the asymptotic behaviour, as $h \to 0$ with $n \to \infty$ and nh constant, is $O(h^p)$ with the decreased exponent a consequence of the accumulation and reinforcement of the errors over n steps. Applying F at the end of the computation gives the error in $F \circ M^n \circ S$, as an approximation to E^n, also equal to $O(h^p)$. This is shown at the right



A completely new way of looking at the dynamics of these multivalue methods (or general linear methods as they are now commonly known) was proposed by D. Stoffer (1993): "General linear methods: connection to one-step methods and invariant curves", *Numer. Math.* **64**, 395–408, as a generalization of the work of U. Kirchgraber (1986): "Multistep methods are essentially one-step methods" *Numer. Math.* **48**, 85–90, who had applied it to the special case of classical linear multistep methods. In the final diagram, S now represents a formal starting method and R a formal one-step method related to M and S so that this diagram commutes. Although there is generally no Runge-Kutta method with the role of R, it is possibly to approximate this formal method using Taylor expansions, written in terms of what are known as B-series.



Even though there is no explicit computational scheme which produces the action of S, the hope is that we can get close to, and remain close to, the manifold on which powers of R evolve, because of attraction. Methods designed on the principle of "Inherent RK stability" seem to be good candidates for efficient integrators and their strong stability property makes them very close in behaviour to Runge-Kutta methods. I would like to understand better if this makes them especially attracted to the invariant manifold defined by their underlying one step methods.

John Butcher, `butcher@math.auckland.ac.nz`