ON THE NUMERICAL INTEGRATION OF ORDINARY DIFFERENTIAL EQUATIONS BY PROCESSED METHODS

S. BLANES[†], F. CASAS[†], AND A. MURUA[‡]

Abstract. We provide a theoretical analysis of the processing technique for the numerical integration of ordinary differential equations. We get the effective order conditions for processed methods in a general setting so that the results obtained can be applied to different types of numerical integrators. We also propose a procedure to approximate the post-processor such that its evaluation is virtually cost free. The analysis is illustrated for a particular class of composition methods.

Key words. Effective order, processing technique, cheap post-processor, initial value problems

AMS subject classifications. 65L05; 65L70; 22E60

1. Introduction.

Given the ordinary differential equation

$$x' = f(x), \qquad x_0 = x(t_0) \in \mathbb{R}^D, \tag{1.1}$$

with $f : \mathbb{R}^D \to \mathbb{R}^D$ and associated vector field (or Lie operator associated with f)

$$F = \sum_{i=1}^{D} f_i(x) \frac{\partial}{\partial x_i}, \tag{1.2}$$

a one-step numerical *integrator* for a time step h, $\psi_h : \mathbb{R}^D \to \mathbb{R}^D$, can be seen as a smooth family of maps with parameter h such that ψ_0 is the identity map. The integrator ψ_h is said to have order of consistency $\geq q$ (or equivalently, to be of order $\geq q$) if

$$\psi_h = \varphi_h + \mathcal{O}(h^{q+1}), \tag{1.3}$$

where φ_h is the h-flow of the ODE (1.1). Then, an approximation to the exact solution $x(h)$ is given by

$$x_h = \psi_h(x_0) = \varphi_h(x_0) + \delta_{h,q}(x_0),$$

where $\delta_{h,q}(x_0) = \mathcal{O}(h^{q+1})$ denotes the local truncation error. The efficiency of the integrator (when compared with methods of the same order and family) depends both on its computational cost and the magnitude of the error term.

In this work we discuss the class of methods obtained by enhancing an integrator ψ_h with processing. The idea of processing can be traced back to the work of Butcher [7] in 1969, where it is considered in the context of Runge–Kutta methods, and is summarized in [12, 19]. Essentially, it consists in obtaining a new (hopefully better) integrator of the form

$$\hat{\psi}_h = \pi_h \circ \psi_h \circ \pi_h^{-1}. \tag{1.4}$$

[†]Departament de Matemàtiques, Universitat Jaume I, 12071-Castellón, Spain (sblanes@mat.uji.es, Fernando.Casas@uji.es). The work of these authors has been partially supported by Fundació Caixa Castelló–Bancaixa. SB has also been supported by Ministerio de Ciencia y Tecnología (Spain) through a contract in the Pogramme Ramón y Cajal 2001 and by the TMR programme through grant EC-12334303730.

[‡]Konputazio Zientziak eta A. A. saila, Informatika Fakultatea, EHU/UPV, Donostia/San Sebastián, Spain (ccpmuura@scsx03.sc.ehu.es).

The method ψ_h is referred to as the *kernel* and the parametric map $\pi_h : \mathbb{R}^D \to \mathbb{R}^D$ as the *post-processor* or *corrector*. Application of n steps of the integrator $\hat{\psi}_h$ leads to

$$\hat{\psi}_h^n = \pi_h \circ \psi_h^n \circ \pi_h^{-1},$$

which can be considered as a change of coordinates in phase space. Thus, it is not required that the kernel ψ_h used to propagate the numerical solution be a *good* integrator. It is sufficient, using dynamical system terminology, that ψ_h be *conjugate* to a good integrator.

Usually one is interested in the case where $\pi_0 = \mathrm{id}$, the identity map, i.e., π_h is also a near-identity map, although it is not intended to approximate the h-flow φ_h. The *pre-processor* π_h^{-1} is applied only once, so that its computational cost may be ignored, then the kernel ψ_h acts once per step and finally the action of the post-processor π_h is evaluated only when output is required. Processing is advantageous if $\hat{\psi}_h$ is a more accurate method than ψ_h and the cost of π_h is negligible: it provides the accuracy of $\hat{\psi}_h$ at the cost of the less accurate method ψ_h.

Although initially intended for Runge–Kutta methods, the processing technique did not become significant in practice, probably due to the difficulties of coupling processing with classical strategies of variable step-sizes. It has been only recently that this idea has proved its usefulness in the context of geometric integration, where constant step-sizes are widely employed.

The aim of geometric integration is to construct numerical schemes for discretizing the differential equation (1.1) whilst preserving certain geometric properties of the vector field F. It is generally recognized that this class of numerical algorithms (the so-called *geometric integrators*) provide a better description of the system (1.1) than standard methods, both with respect to the preservation of invariants and also in the accumulation of numerical errors along the evolution [12, 22].

A typical procedure in geometric integration is to consider one or more low-order methods and compose them with appropriately chosen weights to achieve higher order schemes. The resulting composition method inherits the relevant properties the basic integrator shares with the exact solution, provided these properties are preserved by composition [16].

It has been precisely in this context where the application of processing has proved to be a very powerful tool, allowing to build numerical schemes with both the kernel and the post-processor taken as compositions of basic integrators. In particular, highly efficient processed composition methods have been proposed in the last few years, both in the separable case [3] (including families of Runge–Kutta–Nyström class of methods [5, 14, 15]) and also for slightly perturbed systems [4, 17, 24].

The method ψ_h is of *effective order* q if a post-processor π_h exists for which $\hat{\psi}_h$ is of (conventional) order q [7], that is,

$$\pi_h \circ \psi_h \circ \pi_h^{-1} = \varphi_h + \mathcal{O}(h^{q+1}). \tag{1.5}$$

When analyzing the order conditions $\hat{\psi}_h$ has to verify to be a method of order q, it has been shown that many of them can be satisfied by using π_h [1, 3, 8], so that ψ_h must fulfill a much reduced set of constraints. Furthermore, the error term $\delta_{h,q}(x_0)$ depends on both ψ_h and π_h, and additional conditions can be imposed on the post-processor in order to reduce its magnitude. This allows, on the one hand, to consider kernels involving less evaluations and, on the other hand, to analyze and obtain new and efficient composition methods of high order [5].

In this paper we develop a general theory of the processing technique as applied to the numerical integration of differential equations and derive, under very general assumptions, the conditions to be satisfied by the kernel and the post-processor to attain a given order of consistency. The analysis can be directly applied to different types of numerical methods, including families of composition integrators and Runge–Kutta type methods.

For processed methods whose post-processor is itself constructed as a composition of basic integrators, it turns out that the computational cost of evaluating π_h is usually higher than of ψ_h, so that their use is restricted (in sequential computer environments) to situations where intermediate results are not frequently required. Otherwise the overall efficiency of the methods is highly deteriorated.

Another goal of this work is precisely to show how to avoid this situation, i.e., how to obtain approximations to the post-processor at virtually cost free and without loss of accuracy. The key point is a generalization of a procedure outlined in [14]: π_h is replaced by a new integrator $\hat{\pi}_h \simeq \pi_h$ obtained from the intermediate stages in the computation of ψ_h.

The plan of the paper is as follows. In section 2 we provide a general analysis of processed methods, obtaining the order conditions to be verified by the kernel and the post-processor. In section 3 we propose a cheap alternative for approximating the post-processor, study the corresponding order conditions, and examine the propagation of the error that results from replacing the optimal post-processor by the cheap alternative. Section 4 is concerned with numerical examples and section 5 contains some concluding remarks.

2. Analysis of processed methods.

2.1. Order of consistency of numerical integrators. Let ψ_h be an integrator that approximates the h-flow φ_h of the system (1.1). It is well known that, for each $g \in C^\infty(\mathbb{R}^D, \mathbb{R})$ (i.e, each infinitely differentiable map $g : \mathbb{R}^D \to \mathbb{R}$), $g(\varphi_h(x))$ admits an expansion of the form [21]

$$g(\varphi_h(x)) = \exp(hF)[g](x) = g(x) + \sum_{k\geq 1} \frac{h^k}{k!} F^k[g](x), \qquad x \in \mathbb{R}^D,$$

where F is the vector field (1.2). Let us assume that, for each $g \in C^\infty(\mathbb{R}^D, \mathbb{R})$, $g(\psi_h(x))$ admits an expansion of the form

$$g(\psi_h(x)) = g(x) + h\Psi_1[g](x) + h^2\Psi_2[g](x) + \cdots,$$

where each Ψ_k is a linear differential operator and let Ψ_h denote the series of differential operators

$$\Psi_h = I + \sum_{k\geq 1} h^k \Psi_k$$

so that formally $g \circ \psi_h = \Psi_h[g]$. Clearly, (1.3) is then equivalent to

$$\Psi_k = \frac{1}{k!} F^k, \qquad 1 \leq k \leq q. \tag{2.1}$$

Alternatively, let us consider the series

$$F_h = \log(\Psi_h) = \sum_{m\geq 1} \frac{(-1)^{m+1}}{m} (\Psi_h - I)^m$$

so that, formally, $\Psi_h = \exp(F_h)$ and

$$F_h = \sum_{k\geq 1} h^k F_k, \quad \text{with} \quad F_k = \sum_{m\geq 1} \frac{(-1)^{m+1}}{m} \sum_{j_1+\cdots+j_m=k} \Psi_{j_1} \cdots \Psi_{j_m}. \tag{2.2}$$

It can be shown that the algebraic properties of the linear differential operators Ψ_k imply that such F_h is a series of vector fields. This means the well known fact that the integrator ψ_h can be formally interpreted as the exact 1-flow of the modified vector field F_h [12]. Then, condition (2.1) is equivalent to

$$F_1 = F, \qquad F_k = 0 \quad \text{for} \quad 2 \leq k \leq q. \tag{2.3}$$

It is worth noticing that characterizations (2.1) and (2.3) for the order conditions of the integrator ψ_h are written, in contrast with (1.3), in a way that it is straightforward to extend them to integrators on smooth manifolds, so that we need not to restrict ourselves to integrators on $\mathbb{R}^D$. In fact, the theory of the present paper remains true in a coordinate-free setting, where F_k are vector fields (sections of the tangent bundle) on a finite dimensional smooth manifold.

2.2. Graded Lie algebra of vector fields. We have observed that numerical integrators can be expanded as exponentials of series of vector fields, and these can be used to compare with the exact flow of the system to be integrated numerically. In section 3 we will consider expansions of linear combinations of vector fields, which lie in the associative algebra $\mathcal{B}$ of linear differential operators generated by concatenation of smooth vector fields on $\mathbb{R}^D$, with the identity operator I as the unit element. At this point it seems appropriate to briefly review the main concepts of the theory of Lie algebras in this setting. They will prove to be very useful in the subsequent analysis.

As any associative algebra, the algebra $\mathcal{B}$ has structure of Lie algebra with the commutator $[a, b] = ab - ba$ as the Lie bracket. In other words, the commutator $[a, b]$ is a bilinear operator satisfying

- skew-symmetry: $[a, b] = -[b, a]$;
- the Jacobi identity: $[a, [b, c]] + [b, [c, a]] + [c, [a, b]] = 0$.

The vector fields on $\mathbb{R}^D$ form a subspace of $\mathcal{B}$ that is closed under commutation, i.e., $[F, G]$ is a smooth vector field provided that both F, G are also smooth vector fields.

From (2.2) one has $F_h = hF_1 + h^2F_2 + h^3F_3 + \cdots$, where each F_k is a vector field and h is a symbol that corresponds to the parameter present in the definition of the integrator ψ_h. The set of series of this form inherits a Lie algebra structure from the Lie algebra structure of the set of vector fields if there is a sequence of vector subspaces $\mathcal{L}_k$, $k \geq 1$, of the Lie algebra of vector fields such that $F_k \in \mathcal{L}_k$ for each series $\sum_{k\geq 1} h^k F_k$ and

$$[\mathcal{L}_n, \mathcal{L}_m] \subset \mathcal{L}_{n+m} \qquad \text{for each} \qquad n, m \geq 1. \tag{2.4}$$

In this way the concept of *graded Lie algebra* naturally arises. A graded Lie algebra $\mathcal{L}$ can be defined as a Lie algebra together with a sequence of subspaces $\{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_3, \ldots\}$ of $\mathcal{L}$ such that $\mathcal{L} = \bigoplus_{k\geq 1} \mathcal{L}_k$ and (2.4) holds. The vector spaces $\mathcal{L}_k$ in the graded Lie algebra $\mathcal{L}$ are called *homogeneous components* of $\mathcal{L}$.

Also the notion of *free Lie algebra* is very useful in this setting [23] . Roughly speaking, a Lie algebra $\mathcal{L}$ is free if there exits a set $S \subset \mathcal{L}$ such that (i) any element in $\mathcal{L}$ can be written as a linear combination of nested brackets of elements in S and

(ii) the only linear dependencies among such nested brackets are due to the skew-symmetry property and the Jacobi identity of brackets (see [20] for more details on the theory of free Lie algebras in the context of numerical integration).

Given a Lie algebra $\mathcal{L}$ of vector fields one may consider the associative algebra generated by $\mathcal{L}$ (which is a subalgebra of $\mathcal{B}$). There exists an associative algebra $\mathcal{A} = U(\mathcal{L})$, called the *universal enveloping algebra* [23] of the Lie algebra $\mathcal{L}$ and a unique algebra homomorphism σ of $\mathcal{A}$ onto the algebra of linear differential operators generated by the vector fields in $\mathcal{L}$. That is, any such linear differential operator can be represented as an element of $\mathcal{A}$. In particular, $\sigma(a) = 0$ provided that $a = 0$ for any element $a \in \mathcal{A}$.

The Poincaré–Birkhoff–Witt theorem [23] allows to construct a basis of the universal enveloping algebra $\mathcal{A}$ of $\mathcal{L}$ in terms of a basis of $\mathcal{L}$. More specifically, if $\{L_i\}$ denotes a basis of $\mathcal{L}$, each element of the basis of $\mathcal{A}$ is associated with a family $\{L_{i_1}, \dots, L_{i_k}\}$ of (possibly repeated) elements of the basis of $\mathcal{L}$, and it is the sum of all possible concatenations of basic vector fields $L_{j_1} \cdots L_{j_k}$ such that $(j_1, \dots, j_k)$ is obtained by reordering $(i_1, \dots, i_k)$. When $\mathcal{L}$ is a graded Lie algebra $\mathcal{L} = \bigoplus_{k\geq 1} \mathcal{L}_k$, then $\mathcal{A}$ also admits a graded structure, with $\mathcal{A} = \bigoplus_{k\geq 0} \mathcal{A}_k$, where $\mathcal{A}_0 = \operatorname{span}(I)$ (that is, $\mathcal{A}_n \mathcal{A}_m \subset \mathcal{A}_{n+m}$). Given a basis $\{E_{k,j}\}_{j=1}^{n_k}$ in $\mathcal{L}_k$ for each $k \geq 1$ with $n_k = \dim \mathcal{L}_k$, this procedure leads to a basis $\{D_{k,j}\}_{j=1}^{m_k}$ in $\mathcal{A}_k$ for $k \geq 1$ with $m_k = \dim \mathcal{A}_k$. In particular, this allows obtaining m_k in terms of the dimensions $n_1, \dots, n_k$.

2.3. Effective order conditions. Let us consider now a mapping π_h close to the identity as a post-processor for the integrator ψ_h. Our aim is to obtain characterizations for the order of consistency of the resulting processed integrator (1.4).

As before, let

$$\Pi_h = I + \sum_{k\geq 1} h^k \Pi_k, \qquad \hat{\Psi}_h = I + \sum_{k\geq 1} h^k \hat{\Psi}_k$$

be the series of differential operators such that formally $g \circ \pi_h = \Pi_h[g]$ and $g \circ \hat{\psi}_h = \hat{\Psi}_h[g]$, respectively. Then $\hat{\Psi}_h = \Pi_h^{-1} \Psi_h \Pi_h$, where Π_h^{-1} can be expanded using the same differential operators as in Π_h, and the processed integrator $\hat{\psi}_h$ has order of consistency $\geq q$ if

$$\Psi_h \Pi_h = \Pi_h \exp(hF) + \mathcal{O}(h^{q+1}). \tag{2.5}$$

It is important to notice that different post-processors may result in the same processed integrator, so that it is useful to consider the following definition.

DEFINITION 1. *Two post-processors π_h and $\bar{\pi}_h$ are said to be equivalent with respect to the kernel ψ_h if they give rise to the same processed integrator, i.e., if $\pi_h \circ \psi_h \circ \pi_h^{-1} = \bar{\pi}_h \circ \psi_h \circ \bar{\pi}_h^{-1}$ or, in terms of their respective series of differential operators, if*

$$\Pi_h^{-1} \Psi_h \Pi_h = \bar{\Pi}_h^{-1} \Psi_h \bar{\Pi}_h. \tag{2.6}$$

Remark. Clearly, Π_h and $\bar{\Pi}_h$ are equivalent with respect to the kernel $\Psi_h = \exp(F_h)$ if and only if the vector field $S_h = \log(\Pi_h \bar{\Pi}_h^{-1})$ commutes with F_h, for (2.6) can be written as $\exp(F_h) = \Pi_h \bar{\Pi}_h^{-1} \exp(F_h)(\Pi_h \bar{\Pi}_h^{-1})^{-1}$, or $\exp(F_h)\exp(S_h) = \exp(S_h)\exp(F_h)$ and this is true if and only if $[F_h, S_h] = 0$. In particular, given a post-processor Π_h and a kernel $\Psi_h = \exp(F_h)$, Π_h is equivalent to $\bar{\Pi}_h = \exp(\lambda F_h)\Pi_h$ for an arbitrary $\lambda \in \mathbb{R}$. □

For a given family of integrators $\mathcal{G}$, the effective order conditions are equations on the parameters of the family that indicate the effective order of a particular integrator ψ_h in $\mathcal{G}$. Such effective order conditions can be directly derived from (2.5) for each family of integrators. For instance, for Runge–Kutta methods, (2.5) is equivalent to considering composition of B-series, which is the usual procedure to study the effective order conditions in that setting [8]. However, a general treatment, including the study of the generic number of order conditions, seems difficult with this approach: it would require making specific assumptions on the structure and properties of the series of linear differential operators Ψ_h and Π_h. Instead we propose an alternative based on the vector fields

$$F_h = \sum_{k\geq 1} h^k F_k = \log(\Psi_h), \quad \hat{F}_h = \sum_{k\geq 1} h^k \hat{F}_k = \log(\hat{\Psi}_h), \quad P_h = \sum_{k\geq 1} h^k P_k = \log(\Pi_h).$$

In principle, given a kernel $\Psi_h = \exp(\sum h^k F_k)$, one might look for the best possible post-processor $\Pi_h = \exp(P_h)$ among all possible series of vector fields $P_h = \sum h^k P_k$. However, if F_k is known to belong (for each $k \geq 1$) to a certain Lie algebra $\mathcal{L}$ of vector fields and it is desired that the vector fields $\hat{F}_k$ associated with $\hat{\psi}_h$ also belong to $\mathcal{L}$, then it seems natural to restrict to the case $P_k \in \mathcal{L}$ (this is particularly true if no additional assumptions are made for F_k). We will say that a processed integrator $\hat{\psi}_h$ has order $p \geq q$ in $\mathcal{L}$ if there exist vector fields $P_k \in \mathcal{L}$, $k \geq 1$, such that (2.5) holds with $\Pi_h = \exp(\sum h^k P_k)$.

THEOREM 1. *An integrator ψ_h has effective order $p \geq q$ in $\mathcal{L}$ if and only if there exist vector fields $P_1, \dots, P_{q-1} \in \mathcal{L}$ such that*

$$\begin{aligned} F_1 &= F \\ [P_{k-1}, F] &= F_k + R_k(P_1, \dots, P_{k-2}, F_1, \dots, F_{k-1}), \qquad 1 < k \leq q \end{aligned} \tag{2.7}$$

holds, where

$$R_k = -\sum_{j=1}^{k-2} [P_j, F_{k-j}] + \sum_{l\geq 2} \frac{(-1)^l}{l!} \sum_{j_1+\cdots+j_{l+1}=k} [P_{j_1}, [P_{j_2}, \dots [P_{j_l}, F_{j_{l+1}}] \dots]]. \tag{2.8}$$

Proof. The equality $\hat{\Psi}_h = \Pi_h^{-1} \Psi_h \Pi_h$ can be written in terms of the respective vector fields as $\exp(\hat{F}_h) = \exp(-P_h)\exp(F_h)\exp(P_h)$. Formal application of logarithm in both sides of this expression leads to [23]

$$\hat{F}_h = \exp(-P_h) F_h \exp(P_h) = \exp(\mathrm{ad}_{-P_h}) F_h = \sum_{k=0}^{\infty} \frac{(-1)^k}{k!} \mathrm{ad}_{P_h}^k F_h,$$

where $\mathrm{ad}_A B = [A, B]$. Therefore

$$\hat{F}_h = F_h - [P_h, F_h] + \frac{1}{2!}[P_h, [P_h, F_h]] - \frac{1}{3!}[P_h, [P_h, [P_h, F_h]]] + \cdots,$$

which implies

$$\begin{aligned} \hat{F}_1 &= F_1 \\ \hat{F}_k &= F_k + [F_1, P_{k-1}] + R_k, \qquad k > 1, \end{aligned} \tag{2.9}$$

where $R_2 = 0$, and for $k > 2$, R_k is given by (2.8). Condition (2.5) reads $\hat{F}_1 = F$, $\hat{F}_k = 0$ for $2 \le k \le q$, which is equivalent to (2.7). □

In order to proceed further, we adopt the following assumption.

ASSUMPTION 1. *The kernels ψ_h under consideration in this work are such that their associated vector fields $F_k \in \mathcal{L}_k$, $k \ge 1$, where $\{\mathcal{L}_n\}_{n\ge1}$ is a sequence of subspaces of a certain graded Lie algebra $\mathcal{L}$ of vector fields satisfying (2.4)).*

In typical situations in numerical integration $\mathcal{L}$ is a graded free Lie algebra and $n_k = \dim \mathcal{L}_k$ corresponds to the number of order conditions at order k for non processed methods. The values of n_k, $k \ge 1$, can often be computed by using Witt's formula and their generalizations (see [19] and references therein)

EXAMPLE 1. Let us now consider some particular cases which illustrate Assumption 1 and the context where the results of this paper can be applied.

(1.a) First assume that ODE (1.1) can be written as $x' = f_a(x) + f_b(x)$ and the vector field F is split accordingly as $F = F_a + F_b$. Suppose that the corresponding h-flows $\varphi_h^{[a]}$ and $\varphi_h^{[b]}$ can be exactly computed. Then it is useful to consider numerical integrators of the form

$$\psi_h = \varphi^{[b]}_{\alpha_{2s}h} \circ \varphi^{[a]}_{\alpha_{2s-1}h} \circ \cdots \circ \varphi^{[b]}_{\alpha_2 h} \circ \varphi^{[a]}_{\alpha_1 h}, \tag{2.10}$$

with $\alpha_i \in \mathbb{R}$, i.e., ψ_h is taken as a composition of basic flows. Now Assumption 1 holds for ψ_h with

$$\mathcal{L}_1 = \mathrm{span}(\{F_a, F_b\}), \qquad \mathcal{L}_k = \mathrm{span}\left(\bigcup_{l+m=k} [\mathcal{L}_l, \mathcal{L}_m] \right), \quad k \ge 2. \tag{2.11}$$

If one is interested in obtaining results that are valid for all pairs F_a, F_b of arbitrary vector fields, then one must assume that the only linear dependencies among nested commutators of F_a and F_b can be derived from the skew-symmetry and the Jacobi identity of commutators. In other words, $\mathcal{L} = \bigoplus_{k\ge1} \mathcal{L}_k$ is the graded free Lie algebra generated by the symbolic vector fields F_a, F_b, where both have degree one. In particular, the dimensions n_k of the first homogenous components $\mathcal{L}_k$ are $n_k = 2, 1, 2, 3, 6, 9, 18, 30, 56, 99$.

(1.b) Let us consider the generalized harmonic oscillator with Hamiltonian function

$$H(\mathbf{q}, \mathbf{p}) = \frac{1}{2}\mathbf{p}^T M^{-1} \mathbf{p} + \frac{1}{2}\mathbf{q}^T S \mathbf{q}. \tag{2.12}$$

Here $\mathbf{q}, \mathbf{p} \in \mathbb{R}^d$, M and S are constant symmetric matrices, M invertible. This Hamiltonian (with $S = M^{-1}$) appears in the matrix representation of the time-dependent Schrödinger equation [10], where $\mathbf{q}$ and $\mathbf{p}$ represent the real and imaginary parts of the vector describing the state of the system. With $x = (\mathbf{q}, \mathbf{p})$, $D = 2d$, the corresponding equations of motion can be written as in (1.a) with $f_a(x) = (M^{-1}\mathbf{p}, \mathbf{0})$, $f_b(x) = (\mathbf{0}, -S\mathbf{q})$. Then the Hamiltonian vector field is decomposed as $F = F_a + F_b$, with

$$F_a = \sum_{i=1}^{d} (M^{-1}\mathbf{p})_i \frac{\partial}{\partial q_i}, \qquad F_b = \sum_{i=1}^{d} (-S\mathbf{q})_i \frac{\partial}{\partial p_i}.$$

Now Assumption 1 holds with $\mathcal{L}_k$ given by (2.11). In this case, not all the nested commutators are independent. For instance, $[F_a, [F_a, [F_a, F_b]]] = [F_b, [F_b, [F_b, F_a]]] =$

0. In fact, all nested commutators with an even number of operators F_a, F_b are either zero or a vector field F_C associated with the Hamiltonian $\mathbf{q}^T C \mathbf{p}$, where C is a polynomial matrix function of SM^{-1}. In consequence, $[F_a, F_C]$ is associated with a Hamiltonian function quadratic in $\mathbf{p}$, $[F_a,[F_a,F_C]] = 0$ and similarly $[F_b,[F_b,F_C]] = 0$. In addition, $[F_a,[F_b,F_C]] = [F_b,[F_a,F_C]]$ is also associated with a Hamiltonian function of the form $\mathbf{q}^T C_1 \mathbf{p}$. As a result, $n_{2k} = 1$ and $n_{2k+1} = 2$ for all k.

(1.c) Near-integrable system. It corresponds to the problem $x' = f_a(x) + \varepsilon f_b(x)$ with $|\varepsilon| \ll 1$, which is a particular case of (1.a). The vector field associated with composition (2.10) takes the form $F_h = \sum_{k\geq 1} \sum_{i=1}^{k-1} h^k \varepsilon^i F_{k,i}$, so that we consider a bi-graded Lie algebra with

$$F_a \in \mathcal{L}_{1,0}, \quad F_b \in \mathcal{L}_{1,1}, \quad [\mathcal{L}_{k,i}, \mathcal{L}_{m,j}] \subset \mathcal{L}_{k+m,i+j}, \tag{2.13}$$

and $\mathcal{L}_k = \bigoplus_{i=1}^{k-1} \mathcal{L}_{k,i}$ for $k \geq 2$. We denote $n_{k,i} = \dim \mathcal{L}_{k,i}$, so that obviously $n_k = \sum_{i=1}^{k-1} n_{k,i}$. An explicit formula for the $n_{k,i}$ can be found, in particular, in [16]: for instance, $n_{k,1} = n_{k,k-1} = 1$, $k > 1$, $n_{k,2} = n_{k,k-2} = \left\lfloor \frac{1}{2}(k-1) \right\rfloor$, $k > 2$ and $n_{k,3} = n_{k,k-3} = \left\lfloor \frac{1}{6}(k-1)(k-2) \right\rfloor$, $k > 3$ [19]. Here $\lfloor x \rfloor$ denotes the integer part of x.

(1.d) If $\mathcal{S}_h : \mathbb{R}^D \to \mathbb{R}^D$ is a second order time-symmetric integrator for (1.1), then we can consider integrators of the form [16]

$$\psi_h = \mathcal{S}_{\alpha_s h} \circ \cdots \circ \mathcal{S}_{\alpha_1 h}, \qquad (\alpha_1, \ldots, \alpha_s) \in \mathbb{R}^s. \tag{2.14}$$

It can be shown (see Appendix A) that for such integrators Assumption 1 holds for the graded Lie algebra $\mathcal{L} = \bigoplus_{k\geq 1} \mathcal{L}_k$ generated by certain vector fields $\{Y_1, Y_3, Y_5, \cdots\}$ such that $Y_{2k-1} \in \mathcal{L}_{2k-1}$, $k \geq 1$. The dimensions n_k of the first homogenous components $\mathcal{L}_k$ for $k \geq 1$ are $n_k = 1, 0, 1, 1, 2, 2, 4, 5, 8, 11, 18$ (see, for example, [19, 20]).

(1.e) Runge–Kutta-type methods. The set of rooted trees plays a fundamental role in the standard order theory of Runge–Kutta integrators applied to (1.1) (see for instance [6, 11, 12]). A similar role is played by certain sets of coloured rooted trees in the case of other families of Runge–Kutta-type integrators such as Runge–Kutta–Nyström, partitioned Runge–Kutta, and additive Runge–Kutta methods. Let us generically denote as $\mathcal{T}$ the set of trees corresponding to a family of Runge–Kutta-type integrators, and as $\mathcal{T}_k$ the set of trees in $\mathcal{T}$ with k vertices. For each family of methods, the parameters of any particular qth order integrator must satisfy $n_1 + \cdots + n_q$ algebraic equations, where n_k is the cardinal of $\mathcal{T}_k$. In the standard theory of order conditions, each tree $u \in \mathcal{T}$ is associated to an *elementary differential*, which is a map $F(u) : \mathbb{R}^D \to \mathbb{R}^D$ defined in terms of the map f in (1.1) and its partial derivatives. Now, it can be seen that for each family of Runge–Kutta-type integrators considered above, Assumption 1 holds with

$$\mathcal{L}_k = \operatorname{span}\left(\sum_{i=1}^{D} (F(u))_i \frac{\partial}{\partial y_i} : \ u \in \mathcal{T}_k \right), \quad k \geq 1.$$

The dimensions n_k of the first homogenous components $\mathcal{L}_k$ for $k \geq 1$ are $n_k = 1, 1, 2, 4, 9, 20, 48, 115, 286, 719$ [11]. □

As we have mentioned before, given a kernel of effective order q, the vector fields P_k satisfying (2.7) are not unique. This non-uniqueness is intimately related to the fact that the Lie subalgebra $\mathcal{L}^0 = \{G \in \mathcal{L} : \ [F, G] = 0\}$, i.e., the kernel of ad_F, is non-empty (obviously, $F \in \mathcal{L}^0$). From this perspective, it is useful to choose a direct

complement $\mathcal{L}^*$ of $\mathcal{L}^0$ with respect to $\mathcal{L}$, so that $\mathcal{L}$ is decomposed as a direct sum of two subspaces $\mathcal{L} = \mathcal{L}^0 \oplus \mathcal{L}^*$. For each k, we denote $\mathcal{L}_k^0 = \mathcal{L}^0 \cap \mathcal{L}_k$, $\mathcal{L}_k^* = \mathcal{L}^* \cap \mathcal{L}_k$ and n_k^* the dimension of $\mathcal{L}_k^*$ or, equivalently, $n_k^* = \dim\,[F, \mathcal{L}_k]$, where $[F, \mathcal{L}_k] = [F, \mathcal{L}_k^*]$ is a subspace of $\mathcal{L}_{k+1}$. In general, *if $\mathcal{L}$ is a graded free Lie algebra, then dim $[F, \mathcal{L}_k]$ = dim $\mathcal{L}_k$, $k > 1$, i.e., $n_k^* = n_k$, $k > 1$, and $n_1^* = n_1 - 1$.*

LEMMA 1. *Let $F_k, P_k \in \mathcal{L}_k$ for each $k \geq 1$, with $F_1 = F$. There exist unique $P_k^* \in \mathcal{L}_k^*$, $k \geq 1$, such that the post-processors $\exp(\sum_{k\geq 1} h^k P_k^*)$ and $\exp(\sum_{k\geq 1} h^k P_k)$ are equivalent with respect to the kernel $\Psi_h = \exp(\sum_{k\geq 1} h^k F_k)$.*

Proof. By induction on n, it is sufficient to prove that, if in addition to the assumptions of Lemma 1, $P_1, \ldots, P_{n-1} \in \mathcal{L}^*$ and $P_n \notin \mathcal{L}_n^*$, then there exists a unique $P_n^* \in \mathcal{L}_n^*$ such that $\exp(hP_1 + \cdots + h^{n-1}P_{n-1} + h^n P_n^* + h^{n+1}Q_{n+1} + h^{n+2}Q_{n+2} + \cdots)$ is equivalent to $\exp(\sum h^k P_k)$ with certain $Q_k \in \mathcal{L}_k$, $k > n$.

One first proves that, for arbitrary $P_n^0 \in \mathcal{L}_n^0$, there exists a unique sequence $S_k^* \in \mathcal{L}_k^*$, $k \geq n+1$ such that $S_h = -h^n P_n^0 + \sum_{k\geq n+1} h^k S_k^*$ commutes with F_h. One considers $P_n^0 \in \mathcal{L}_n^0$, $P_n^* \in \mathcal{L}_n^*$ such that $P_n = P_n^0 + P_n^*$, and observe that, by choosing S_h as above, $\exp(\sum h^k P_k)$ is equivalent to

$$\exp\left(-h^n P_n^0 + \sum_{k\geq n+1} h^k S_k^*\right) \exp\left(\sum_{k\geq 1} h^k P_k\right) = \exp\left(\sum_{k=1}^{n-1} h^k P_k + h^n P_n^* + \cdots\right).$$

The uniqueness of P_n^* directly follows from (2.9). ∎

In other words, Lemma 1 shows that we can take into account only post-processors such that $P_k \in \mathcal{L}_k^*$ without restricting the choice of the processed integrator. In addition, ψ_h has effective order $p \geq q$ in $\mathcal{L}$ if and only if there exist vector fields $P_k \in \mathcal{L}_k^*$, $k \leq q-1$, such that (2.7) hold. Moreover, such vector fields are unique in $\mathcal{L}^*$.

On the other hand, equations (2.9) lead directly to the following result.

LEMMA 2. *If the vector fields F_k, $\hat{F}_k \in \mathcal{L}_k$, $P_k \in \mathcal{L}_k^*$, $k \geq 1$ are associated with the kernel ψ_h, the processed method $\hat{\psi}_h$ and the post-processor π_h, respectively, it follows that*

(a) if ψ_h is a method of order d then $\pi_h = \mathrm{id} + \mathcal{O}(h^d)$ or equivalently

$$F_k = \hat{F}_k = 0, \quad 2 \leq k \leq d \quad \Longrightarrow \quad P_k = 0, \quad 1 \leq k \leq d-1;$$

(b) provided the kernel is such that $\psi_{-h} = \psi_h^{-1} + \mathcal{O}(h^{2d+2})$ then it holds that $\hat{\psi}_{-h} = \hat{\psi}_h^{-1} + \mathcal{O}(h^{2d+2})$ if and only if $\pi_{-h} = \pi_h + \mathcal{O}(h^{2d+1})$. In terms of vector fields,

$$F_{2k} = \hat{F}_{2k} = 0, \quad 1 \leq k \leq d \quad \Longleftrightarrow \quad F_{2k} = P_{2k-1} = 0, \quad 1 \leq k \leq d.$$

Next we rewrite the order conditions (2.7) for the processed integrator as a system of (polynomial) equations in the coefficients of the vector fields F_k in a basis $\{E_{k,i}\}_{i=1}^{n_k}$ of $\mathcal{L}_k$, $k \geq 1$. Such conditions take a very simple form if the basis of $\mathcal{L}_{k+1}$ ($k \geq 1$) includes a basis of $[F, \mathcal{L}_k] = [F, \mathcal{L}_k^*]$. This can be done, for instance, as follows. First choose a basis $\{E_{k,i}^*\}_{i=1}^{n_k^*}$ of $\mathcal{L}_k^*$ (of course, such a basis of $\mathcal{L}_k^*$ can always be chosen as a subset of the basis $\{E_{k,i}\}_{i=1}^{n_k}$ of $\mathcal{L}_k$). Then, take

$$E_{k+1,n_{k+1}-n_k^*+i} = [F, E_{k,i}^*], \quad \text{for} \quad i = 1, \ldots, n_k^*, \tag{2.15}$$

and complete the basis of $\mathcal{L}_{k+1}$ by choosing $n_{k+1} - n_k^*$ elements of $\mathcal{L}_{k+1}$, say $E_{k+1,i}$, $i = 1, \ldots, n_{k+1} - n_k^*$, such that $\{E_{k+1,i}\}_{i=1}^{n_{k+1}}$ spans $\mathcal{L}_{k+1}$.

From now on, we assume that the basis of $\mathcal{L}_k$ and $\mathcal{L}_k^*$ have been constructed in such a way that (2.15) holds. Let us write

$$F_k = \sum_{i=1}^{n_k} f_{k,i} E_{k,i}, \qquad R_k = \sum_{i=1}^{n_k} r_{k,i} E_{k,i}, \qquad P_{k-1} = \sum_{i=1}^{n_{k-1}^*} p_{k-1,i} E_{k-1,i}^*. \tag{2.16}$$

The effective order conditions (2.7) are then expressed in terms of the coefficients $f_{k,i}$, $r_{k,i}$ and $p_{k-1,i}$ as follows.

THEOREM 2. *The scheme ψ_h, satisfying Assumption 1, has effective order $p \geq q$ if and only if*

$$f_{k,i} = -r_{k,i}, \qquad 1 \leq i \leq l_k := n_k - n_{k-1}^*, \tag{2.17}$$

$$p_{k-1,i} = -f_{k,l_k+i} - r_{k,l_k+i}, \qquad 1 \leq i \leq n_{k-1}^*, \tag{2.18}$$

for $1 < k \leq q$. If, in addition, ψ_h is time-symmetric (i.e., if $\psi_{-h} \circ \psi_h = \mathrm{id}$) then, for even values of k, conditions (2.17) are automatically satisfied and equations (2.18) reduce to $p_{k-1,i} = 0$.

Proof. Assumption 1 implies that each R_k in (2.9) belongs to $\mathcal{L}_k$ and thus expressions (2.16) hold, where each $r_{k,i}$ is a polynomial in the coefficients $f_{l,j}$, $p_{l-1,j}$, $l = 2, \ldots, k-1$ (as R_k in (2.9) is a Lie polynomial in F_l, P_{l-1}, $l \leq k-1$). Conditions (2.7) together with (2.15) then lead to (2.17) and (2.18).

In the particular case of a time-symmetric kernel, then $F_{2i} = 0$. The conclusion readily follows from Lemma 2. □

COROLLARY 2.1. *A total number of*

$$s(q) \equiv \sum_{k=1}^{q} n_k - \sum_{k=1}^{q-1} n_k^* = n_q + \sum_{k=1}^{q-1} (n_k - n_k^*)$$

conditions have to be satisfied by a given kernel ψ_h of effective order $p \geq q > 1$. If $\mathcal{L}$ is a graded free Lie algebra, this number is $s(q) = n_q + 1$. If ψ_h is time-symmetric, then the total number of effective order conditions reduces to $\bar{s}(2) = n_1$ and $\bar{s}(2n) = \sum_{k=2}^{n} (n_{2k-1} - n_{2k-2}^*)$.

Proof. Equations (2.18) hold for any kernel provided the post-processor is appropriately chosen, and equations (2.17) give $l_k = n_k - n_{k-1}^*$ conditions for each $k = 2, \ldots, q$. This, together with the n_1 consistency conditions corresponding to $F_1 = F$, leads to $s(q)$ equations on the coefficients $f_{k,i}$. On the other hand, if the graded Lie algebra is free then $n_k^* = n_k$, $k > 1$, and $n_1^* = n_1 - 1$. Finally, if ψ_h is time-symmetric one has to count only the number of conditions for odd values of k. □

Remark. For any kernel, each $r_{k,i}$ is a polynomial in $f_{l,j}$, $p_{l-1,j}$, $l = 2, \ldots, k-1$. Recursive substitution of (2.17)–(2.18) in such polynomials $r_{k,i}$ leads to an equivalent system of equations of the form (2.17)–(2.18), where now each $r_{k,i}$ is a polynomial in the coefficients $f_{l,j}$, $l = 2, \ldots, k-1$, $j = n_l - n_{l-1}^* + 1, \ldots, n_l$. □

EXAMPLE 2. Next we provide the total number of order conditions for the particular cases collected in Example 1.

(2.a) For the composition methods of Example (1.a), $\mathcal{L}^0 = \mathrm{span}(\{F\})$. Therefore $n_1^* = n_1 - 1 = 1$, and among the different choices for $\mathcal{L}_1^*$, one can take for instance $\mathcal{L}_1^* = \mathrm{span}(\{F_a\})$, $\mathcal{L}_1^* = \mathrm{span}(\{F_b\})$, or $\mathcal{L}_1^* = \mathrm{span}(\{F_a - F_b\})$. For each $k \geq 2$, $\mathcal{L}_k \cap \mathcal{L}^0 = \{\emptyset\}$, so that $\mathcal{L}_k^* = \mathcal{L}_k$, $n_k^* = n_k$, and one can choose $E_{k,i}^* = E_{k,i}$. According to Corollary 2.1, the total number of effective order conditions is then $s(q) = n_q + 1$, a result already obtained in [3].

(2.b) For the harmonic oscillator (2.12) considered in Example (1.b), the number of effective order conditions $s(q)$ is substantially reduced. As we have seen, $n_{2k-1} = 2$ and $n_{2k} = 1$ for each $k \geq 1$. The basis elements can recursively be built, for example, as follows: $E_{1,1} = F = F_a + F_b$, $E_{1,2} = F_a - F_b$, and for $k \geq 1$, $E_{2k,1} = [F, E_{2k-1,2}]$, $E_{2k+1,1} = [F_a - F_b, E_{2k,1}]$, $E_{2k+1,2} = [F, E_{2k,1}]$, with $\mathcal{L}_{2k} = \mathrm{span}(\{E_{2k,1}\})$ and $\mathcal{L}_{2k+1} = \mathrm{span}(\{E_{2k+1,1}, E_{2k+1,2}\})$. From Example (1.b), we have that $[F_a, [F_a, E_{2k,1}]] = [F_b, [F_b, E_{2k,1}]] = 0$ and $[F, E_{2k+1,1}] = -[F, E_{2k+1,2}]$, so that

$$n^*_{2k} = \dim\, [F, \mathcal{L}_{2k}] = \dim\, \mathrm{span}\,(\{[F, E_{2k,1}]\}) = 1$$
$$n^*_{2k+1} = \dim\, [F, \mathcal{L}_{2k+1}] = \dim\, \mathrm{span}\,(\{[F, E_{2k+1,1}], [F, E_{2k+1,2}]\}) = 1,$$

i.e., $n^*_k = 1$ for all k, and thus, $s(q) = \lfloor (q+1)/2 \rfloor + 1$ (or $s(2n-1) = s(2n) = n+1$). Counting the number of effective order conditions and the number of variables from the composition (2.10) we observe that, if the equations have real solutions, in principle methods of effective order $4s-2$ can be obtained. Furthermore, an interesting feature of schemes (2.10) applied to the generalized harmonic oscillator (2.12) is that for any kernel of the form (2.10), a post-processor exists such that the processed integrator is time-symmetric. This is a consequence of the fact that $l_{2k} := n_{2k} - n^*_{2k-1} = 0$ for all k, and therefore $\hat{F}_{2k} = 0$ if the post-processor is appropriately chosen (i.e., if $p_{2k-1,1} = -f_{2k,2} - r_{2k,2}$).

(2.c) Since the near-integrable problem is a particular case of (1.a), we can build a basis of $\mathcal{L}_k$ and then, by taking into account that $\mathcal{L}_k = \bigoplus_{i=1}^{k-1} \mathcal{L}_{k,i}$, obtain a basis of each $\mathcal{L}_{k,i}$. According to (2.a), $\mathcal{L}_k = \mathcal{L}^*_k$ and $\mathcal{L}_{k,i} = \mathcal{L}^*_{k,i}$ for $k > 1$, $i = 1, \ldots, k-1$. If we take $\mathcal{L}^0_1 = \mathrm{span}(\{F_a\})$ then $n_{1,0} = n_{1,1} = 1$ and $n^*_{1,0} = 0$, $n^*_{1,1} = 1$.

Usually, one is interested in designing methods such that [4]

$$F_h - F = \mathcal{O}(\varepsilon h^{s_1+1} + \varepsilon^2 h^{s_2+1} + \varepsilon^3 h^{s_3+1} + \cdots). \tag{2.19}$$

A method which satisfies this condition is said to be of order $(s_1, s_2, s_3, \ldots, s_q = q)$. We are interested in the case where $s_i \geq s_{i+1}$ and the list terminates with $\varepsilon^q h^{q+1}$, q being the standard order of consistency of the method. Observe that s_1 is the order of consistency the method would have in the limit $\varepsilon \to 0$.

To count the number of order conditions one has to consider each power of ε separately. In a non-processed method this number is $n_{1,0} + n_{1,1} + \sum_{i=1}^{q-1} \sum_{k=i+1}^{s_i} n_{k,i}$, whereas in the processed case this number reduces to (applying Corollary 2.1 to each power of ε separately)

$$s(s_1, \ldots, q) = 1 + \sum_{i=1}^{q-1} n_{s_i,i}.$$

Since $n_{s_1,1} = 1$, the number of order conditions is independent of s_1, and $(s_1, 2)$ methods can be obtained just with a consistent kernel (a first order method) [24]. If $s_1 = \cdots = s_q = q$ the result of Corollary 2.1 is recovered.

(2.d) For kernels constructed as compositions of a basic 2nd order symmetric integrator (2.14), then $\mathcal{L}^0 = \mathcal{L}_1 = \mathrm{span}(\{F\})$. Whence $n^*_1 = n_1 - 1 = 0$, and for each $k \geq 2$, $\mathcal{L}_k \cap \mathcal{L}^0 = \{\emptyset\}$, so that $\mathcal{L}^*_k = \mathcal{L}_k$, $n^*_k = n_k$. The total number of effective order conditions is then $s(q) = n_q + 1$.

(2.e) For the family of Runge–Kutta methods, the situation is very similar to (2.d). Now $n_1 = 1$, $n^*_1 = 0$, and $n^*_k = n_k$ for $k \geq 2$, and thus, the number of

conditions to have effective order conditions q is $s(q) = n_q + 1$, that is, the number of rooted trees with q vertices plus one. This result was obtained by Butcher and Sanz-Serna in [8]. As for Runge–Kutta–Nyström methods, the situation is similar to (2.a), with $n_1 = 1$, $n_1^* = 0$, and $n_k^* = n_k$ for $k \geq 2$, and Corollary 2.1 again leads to $s(q) = n_q + 1$. □

For a kernel of effective order q (i.e., satisfying equations (2.17) for $k \leq q$ but not for $k = q+1$), one could in principle determine a post-processor such that (2.18) *holds also for all* $k > q$. From now on we shall refer to that post-processor as *optimal*, as it causes many terms of each $\hat{F}_k = \sum_{i=1}^{n_k} \hat{f}_{k,i} E_{k,i}$ of the processed method $\hat{\psi}_h$ to cancel ($\hat{f}_{k,i} = 0$ for $i = n_k - n_{k-1}^* + 1, \ldots, n_k$).

Remark. This optimal post-processor is not uniquely defined, and it depends on the way a basis of $[F, \mathcal{L}_{k-1}]$ ($k \geq 2$) is completed to get a basis of $\mathcal{L}_k$ (i.e., on the choice of the direct complement $\bar{\mathcal{L}}_k := \mathrm{span}(\{E_{k,i}\}_{i=1}^{n_k - n_{k-1}^*})$ of $[F, \mathcal{L}_{k-1}]$ with respect to $\mathcal{L}_k$). In fact, we are determining the optimal P_h by requiring that the vector field $\hat{F}_h$ belongs to $\bar{\mathcal{L}} := \bigoplus_{k \geq 1} \bar{\mathcal{L}}_k$ (i.e., that the projection onto $[F, \mathcal{L}]$ parallel to $\bar{\mathcal{L}}$ is canceled). This obviously depends on the choice of $\bar{\mathcal{L}}$. We will nevertheless still use the term 'optimal post-processor' by implicitly assuming that this refers to a prescribed decomposition $\mathcal{L} = \bar{\mathcal{L}} \oplus [F, \mathcal{L}]$. □

DEFINITION 2. *We denote by* $\mathbb{P}_k$ *the set of maps* $\pi_h : \mathbb{R}^D \to \mathbb{R}^D$ *whose Taylor expansion is identical to the optimal post-processor up to order* k *(i.e., their difference is* $\mathcal{O}(h^{k+1})$*).*

Thus, we have a qth-order processed integrator $\hat{\psi}_h$ if the kernel ψ_h has effective order q and the post-processor π_h is in $\mathbb{P}_{q-1}$. If in addition $\pi_h \in \mathbb{P}_q$, then the leading term of the resulting vector field $\hat{F}_h - hF$ coincides with the leading term of the optimal post-processor.

3. Cheap post-processing. In most cases the optimal post-processor can be accurately approximated, but it usually turns into a scheme which is (at least) as expensive to evaluate as the kernel. Since the pre-processor is evaluated only once, it makes sense to use this (typically) expensive approximation. On the contrary, using the more accurate approximation to the post-processor for obtaining intermediate results along the numerical integration process may deteriorate the efficiency of the method, especially if output is frequently required as occurs, for instance, in the calculation of Lyapunov exponents and the computation of Poincaré maps in dynamical systems. It is then reasonable to look for an approximation $\hat{\pi}_h$ to the optimal post-processor as cheap to compute as possible. Usually, such a cheap post-processor $\hat{\pi}_h$ will be a less accurate approximation to the optimal post-processor, but the error $\hat{\pi}_h(y_n) - \pi_h(y_n)$ thus introduced will not be propagated: as we shall see in section 3.2, such an error eventually is overtaken by the global error of the underlying processed integrator in typical situations (where the global error grows at least linearly in time).

Computationally cheap approximations to the optimal post-processor can be obtained by applying different techniques. Here we present an approach which can be considered cost free. In essence, π_h is approximated by reusing intermediate calculations obtained in the evaluation of the kernel ψ_h.

More precisely, let $x(t_0) = x_0$ be the initial value of the problem, and $y_n = \psi_h^n(\pi_h^{-1}(x_0))$. Then we approximate $x_n = \pi_h(y_n)$ as the linear combination

$$x_n \approx \sum_{i=-s}^{s} w_i Y_i \tag{3.1}$$

of intermediate values $Y_i \in \mathbb{R}^D$ computed when evaluating $y_n = \psi_h(y_{n-1})$ and $y_{n+1} = \psi_h(y_n)$. Here we only consider intermediate values from two steps, although more could also be used. There is no loss of generality though, since using $2m$ steps is equivalent to using two steps of the kernel ψ_h^m.

To proceed further, the existence of such intermediate values has to be guaranteed.

ASSUMPTION 2. *After evaluating $y_{n+1} = \psi_h(y_n)$ with a kernel ψ_h satisfying Assumption 1, the intermediate values Y_i, $i = 1, \dots, s$, are available. These can be interpreted as $Y_i = \phi_h^{(i)}(y_n)$ for suitable integrators $\phi_h^{(i)}$ satisfying Assumption 1.*

3.1. Conditions on the cheap post-processor. Under Assumption 2, we consider (3.1) with $Y_0 = y_n$, and $Y_{-i} = \phi_h^{(s-i)}(y_{n-1})$, $i = 1, \dots, s$, that is, $Y_{-i} = \phi_h^{(-i)}(y_n)$, where $\phi_h^{(-i)} = \phi_h^{(s-i)} \circ \psi_h^{-1}$. Thus, (3.1) can be rewritten as

$$x_n \approx \hat{\pi}_h(y_n), \qquad \text{where} \qquad \hat{\pi}_h(y) = \sum_{i=-s}^{s} w_i \phi_h^{(i)}(y) \tag{3.2}$$

and each $\phi_h^{(i)}(y)$, $-s \le i \le s$, is an integrator satisfying Assumption 1.

EXAMPLE 3. We illustrate Assumption 2 in some particular cases.

(3.a) For kernels of the form (2.10), Assumption 2 holds for the intermediate values $Y_j = \phi_h^{(j)}(y_n)$ ($-2s \le j \le 2s$, because we have $2s$ intermediate stages per step), where

$$Y_{2i-1} = \varphi^{[a]}_{\alpha_{2i-1}h} \circ \cdots \circ \varphi^{[a]}_{\alpha_1 h}(y_n), \quad Y_{2i} = \varphi^{[b]}_{\alpha_{2i}h} \circ \cdots \circ \varphi^{[a]}_{\alpha_1 h}(y_n),$$
$$Y_{-2i+1} = \varphi^{[a]}_{\alpha_{2s-2i+1}h} \circ \cdots \circ \varphi^{[a]}_{\alpha_1 h}(y_{n-1}), \quad Y_{-2i} = \varphi^{[b]}_{\alpha_{2s-2i}h} \circ \cdots \circ \varphi^{[a]}_{\alpha_1 h}(y_{n-1})$$

and $-s \le i \le s$.

(3.b) For kernels of the form (2.14), Assumption 2 holds for the intermediate values $Y_i = \phi_h^{(i)}(y_n)$ ($-s \le i \le s$), where

$$Y_i = \mathcal{S}_{\alpha_i h} \circ \cdots \circ \mathcal{S}_{\alpha_1 h}(y_n), \quad Y_{-i} = \mathcal{S}_{\alpha_{s-i} h} \circ \cdots \circ \mathcal{S}_{\alpha_1 h}(y_{n-1}). \tag{3.3}$$

(3.c) Recall that a Runge–Kutta integrator ψ_h for the system (1.1) reads

$$\psi_h(y) = y + h \sum_{i=1}^{s} b_i f(Y_i), \quad Y_i = y + h \sum_{j=1}^{s} a_{ij} f(Y_j), \quad i = 1, \dots s, \tag{3.4}$$

where b_i, a_{ij} are parameters of the method. Clearly, Assumption 2 holds for the intermediate stages Y_i ($1 \le i \le s$), since each Y_i defines a Runge–Kutta scheme. The internal stages of other Runge-Kutta–type families of integrators can be similarly seen to satisfy Assumption 2. □

Next we study the conditions the coefficients w_i must satisfy so that $\hat{\pi}_h \in \mathbb{P}_l$ with l as high as possible. In fact, this is guaranteed for a given $l \ge 1$ if

$$\hat{\Pi}_h := \sum_{i=-s}^{s} w_i \Phi_h^{(i)} = \Pi_h + \mathcal{O}(h^{l+1}), \tag{3.5}$$

where $\Phi_h^{(i)}$ ($-s < i \le s$) is the series of differential operators $\Phi_h^{(i)} = I + \sum_{j \ge 1} h^j \Phi_j^{(i)}$ such that formally, $g \circ \phi_h^{(i)} = \Phi_h^{(i)}[g]$. From Assumption 2 we have that $\Phi_h^{(i)} = \exp(F_h^{(i)})$, where $F_h^{(i)} = \sum_{k \ge 1} h^k F_k^{(i)}$ and $F_k^{(i)} \in \mathcal{L}_k$, $k \ge 1$.

Observe that $\hat{\pi}_h$ cannot be interpreted as the exact 1-flow of a formal vector field in the Lie algebra $\mathcal{L}$, that is, $\log(\hat{\Pi}_h) \notin \mathcal{L}$. However, since $\hat{\Pi}_h$ is defined as a linear combination of exponentials of (formal) vector fields in $\mathcal{L}$, it is clear that $\hat{\Pi}_h$ is a formal series of elements in the associative algebra of linear differential operators generated by the vector fields in $\mathcal{L}$, and therefore, as noted in subsection 2.2, the series $\hat{\Pi}_h$ can be appropriately represented by using the universal enveloping algebra $\mathcal{A}$ of $\mathcal{L}$.

According to that discussion, Π_h and each Φ_h^i (hence $\hat{\Pi}_h$) can be expressed as

$$\Pi_k = \sum_{j=1}^{m_k} \pi_{k,j} D_{k,j}, \qquad \Phi_k^{(i)} = \sum_{j=1}^{m_k} \phi_{k,j}^{(i)} D_{k,j}, \qquad -s \le i \le s, \tag{3.6}$$

where $\pi_{k,j}, \phi_{k,j}^{(i)} \in \mathbb{R}$, and $\{D_{k,j}\}_{j=1}^{m_k}$ is a basis of the kth homogeneous component $\mathcal{A}_k$ of $\mathcal{A}$, constructed, for instance, at the end of subsection 2.2. In particular, since $\Phi_h^{(i)} = \exp(F_h^{(i)})$ with $F_h^{(i)} = \sum_{k\ge 1} h^k F_k^{(i)}$, $F_k^{(i)} = \sum_{j=1}^{n_k} f_{k,j}^{(i)} E_{k,j}$, we have that each $\phi_{k,j}^{(i)}$ in (3.6) is a polynomial function of $f_{l,r}^{(i)}$, $l \le k$, $r \le n_l$. The same is true for the coefficients $\pi_{k,j}$ and the coefficients $p_{k,j}$ in $P_k = \sum_{j=1}^{n_k} p_{k,j} E_{k,j}$. Hence, (3.5) is equivalent to a system of linear equations on the unknowns w_i, i.e.,

$$\sum_{i=-s}^{s} w_i \phi_{k,j}^{(i)} = \pi_{k,j}, \qquad 1 \le j \le m_k, \qquad 0 \le k \le l. \tag{3.7}$$

In particular, $\hat{\pi}_h \in \mathbb{P}_0$ is equivalent to $\sum_{i=-s}^{s} w_i = 1$, and the number of equations (3.7) required for $\hat{\pi}_h \in \mathbb{P}_l$ is then $1 + m_1 + \cdots + m_l$.

When the number of unknowns w_i in (3.1) is larger than the number of equations (3.7) required so that $\hat{\pi}_h \in \mathbb{P}_l$ for a given l, then one can use this freedom to minimize the difference with the optimal post-processor at higher orders.

3.1.1. Cheap post-processing for time-symmetric kernels. In the important case of time-symmetric kernels, then $\Pi_k = 0$ for odd indices k. In addition, it is typically the case that $\Phi_h^{(-i)} = \Phi_{-h}^{(i)}$ for $-s \le i \le s$. The choice $w_{-i} = w_i$ for all i in (3.1) then makes sense, that is,

$$\hat{\pi}_h = w_0 \,\mathrm{id} + \sum_{i=1}^{s} w_i \,(\phi_h^{(i)} + \phi_h^{(-i)}), \tag{3.8}$$

so that $(w_0 = 1 - 2\sum_{i=1}^{s} w_i)$

$$\hat{\Pi}_h = w_0\, I + \sum_{i=1}^{s} w_i (\Phi_h^{(i)} + \Phi_{-h}^{(i)}) = I + 2 \sum_{r\ge 1} h^{2r} \left(\sum_{i=1}^{s} w_i \Phi_{2r}^{(i)} \right).$$

This guarantees that equations (3.7) are automatically satisfied for odd values of k, and the equations for even values of k are of the form

$$2 \sum_{i=1}^{s} w_i \phi_{k,j}^{(i)} = \pi_{k,j}, \qquad 1 \le j \le m_k. \tag{3.9}$$

Hence, the number of equations that remain to be satisfied by the unknowns $w_1, \ldots, w_s$ so that $\hat{\pi}_h \in \mathbb{P}_{2r-1}$ is $m_2 + \cdots + m_{2r-2}$.

EXAMPLE 4. A kernel of the form (2.14) is time symmetric if $\alpha_{s-i+1} = \alpha_i$ for each i. We already know that, in that case, $f_{2i,j} = 0$, $p_{2i-1,j} = 0$. In addition, one has $\phi_h^{(-i)} = \phi_{-h}^{(i)}$ for the intermediate values (3.3) to be used for the cheap post-processor. Hence, we take $w_{-i} = w_i$ $(1 \le i \le s)$ in (3.2). Thus, in particular, a total number of $m_2 + m_4 = 1 + 3 = 4$ linear equations (3.9) have to be satisfied in order that $\hat{\pi}_h \in \mathbb{P}_5$. In Appendix A these equations are written explicitly in terms of the coefficients α_i of the kernel. □

3.1.2. Improved specialized cheap post-processors. As we have seen, condition (3.5) is sufficient for a cheap post-processor (3.2) to belong to $\mathbb{P}_l$. However, this is not necessary in general. In fact, (3.5) means that

$$\sum_{i=-s}^{s} w_i g(\phi^{(i)}(y)) = g(\pi_h(y)) + \mathcal{O}(h^{l+1}), \tag{3.10}$$

for any $g \in C^\infty(\mathbb{R}^D, \mathbb{R})$, $y \in \mathbb{R}^D$, but in order that $\hat{\pi}_h \in \mathbb{P}_l$, (3.10) has to be imposed only for $g = g_j$, $j = 1, \dots, D$, where g_j is the projection onto the jth coordinate. As we will see, this observation leads in certain cases to a reduction in the number of conditions required.

EXAMPLE 5. Consider again the family of Runge–Kutta schemes (3.4). Recall that in that case, n_k is the number of rooted trees with k vertices, and it is not difficult to show that $m_k = n_{k+1}$ for each $k \ge 1$. Now, the integrator $\hat{\pi}_h$ (3.2) is itself a Runge–Kutta method (provided that $\sum w_i = 1$) and standard Runge–Kutta theory can be used to show that $1 + n_1 + \cdots + n_l$ conditions on the parameters w_i are sufficient for $\hat{\pi}_h \in \mathbb{P}_l$, instead of the $1 + m_1 + \cdots + m_l = 1 + n_1 + \cdots + n_{l+1}$ conditions obtained from (3.5). □

One could also consider the use of cheap post-processors with different sets of values of the parameters w_i for different components of y. In that case, one only needs to impose (3.10) for the projection onto the corresponding component. Under certain assumptions, this also leads to a reduction in the number of conditions to be satisfied by the coefficients w_i. To be more specific, let us consider the following assumptions.

ASSUMPTION 3. *For a certain j, there exists $r_j \in C^\infty(\mathbb{R}^D, \mathbb{R})$ such that for any $k \ge 1$ and $\Phi_k \in \mathcal{A}_k$, $\Phi_k[g_j]$ can be written as a linear combination of elements in $\mathcal{A}_{k-1}$ acting on r_j.*

One can show that, under Assumption 3, $2 + (m_1 + \cdots + m_{l-1})$ conditions on the parameters w_i guarantee that (3.10) holds for $g = g_j$ (such conditions are independent of the actual function r_j).

Assumption 3 holds, in particular, for every component for Runge–Kutta methods, that is, for the graded free Lie algebra associated with the set of rooted trees considered in Example (1.e). It also holds for the case of integrators in Example (1.d), provided the basic 2nd order symmetric method $\mathcal{S}_h$ is the implicit trapezoidal rule.

ASSUMPTION 4. *For a certain j, there exists $r_j \in C^\infty(\mathbb{R}^D, \mathbb{R})$ such that for any $k \ge 2$ and $\Phi_k \in \mathcal{A}_k$, $\Phi_k[g_j]$ can be written as a linear combination of elements in $\mathcal{A}_{k-2}$ acting on r_j.*

In a similar way, it can be shown that, under Assumption 4, $1 + m_1 + (1 + m_1 + \cdots + m_{l-2})$ conditions on the parameters w_i are sufficient for (3.10) to hold with $g = g_j$.

It can be seen that, when $\mathcal{L}$ is the Lie algebra corresponding to Runge–Kutta–Nyström methods (Example (1.e)), then Assumption 4 holds for the components

corresponding to positions, while Assumption 3 holds for the velocity components.

For the case of integrators in Example (1.d), if the basic 2nd order symmetric method $\mathcal{S}_h$ is the Störmer–Verlet method, then again Assumption 3 holds for velocities, while Assumption 4 holds for positions.

3.2. Error propagation. Our purpose now is to analyze the propagation of the global error when the post-processor is approximated by the linear combination $\hat{\pi}_h$ of intermediate values obtained in the computation of the kernel. As a general rule, the precision of the final results is not conditioned by the use of a very accurate post-processor, whereas the error introduced by replacing the pre-processor π_h^{-1} by $\hat{\pi}_h^{-1}$ can grow significantly along the integration.

To justify this assertion, let us consider a post-processor π_h in $\mathbb{P}_l$, with $l \geq q$, and q is the order of the processed integrator $\hat{\psi}_h$. After n steps we have

$$x_n = \hat{\psi}_h^n(x_0) = \pi_h \circ \psi_h^n \circ \pi_h^{-1}(x_0) = x(t_n) + e_{h,q}(n, x_0),$$

where $t_n = nh$ and $e_{h,q}(n, x_0)$ is the global error of the method. If $\hat{\pi}_h \in \mathbb{P}_k$, with $k < q$, is used as the post-processor, then

$$\tilde{x}_n \equiv \hat{\pi}_h \circ \psi_h^n \circ \pi_h^{-1}(x_0) = \hat{\pi}_h \circ \pi_h^{-1} \circ \hat{\psi}_h^n(x_0) = x(t_n) + e_{h,q}(n, x_0) + \hat{\delta}_{h,k}(x_n).$$

Here $\hat{\delta}_{h,k} \equiv \hat{\pi}_h \circ \pi_h^{-1} - \mathrm{id} = \mathcal{O}(h^{k+1})$ is an error of local nature, which in general can be bounded independently of n, while the global error typically grows as n increases. On the other hand, if $\hat{\pi}_h^{-1}$ is used as pre-processor then

$$\begin{aligned}\hat{x}_n &\equiv \pi_h \circ \psi_h^n \circ \hat{\pi}_h^{-1}(x_0) = \hat{\psi}_h^n \circ \pi_h \circ \hat{\pi}_h^{-1}(x_0) = \hat{\psi}_h^n(x_0 + \tilde{\delta}_{h,k}(x_0)) \\ &= x(t_n) + e_{h,q}(n, x_0) + \tilde{e}_{h,k}(n, x_0),\end{aligned}$$

where $\tilde{e}_{h,k}$ corresponds to the propagation of the initial error $\tilde{\delta}_{h,k} \equiv \pi_h \circ \hat{\pi}_h^{-1} - \mathrm{id} = \mathcal{O}(h^{k+1})$. Now the error term $\tilde{e}_{h,k}$ is not of local character, and can grow significantly as n increases.

It is important to notice that, when the kernel approximately preserves an integral of motion I, and $\hat{\pi}_h$ is used as post-processor, the accuracy in the value of I can be reduced. Nevertheless, one must keep in mind that this corresponds to a local error which is not propagated and that, if required, one can always use a more precise approximation to the post-processor at selected times.

4. Numerical Experiments. In this section we examine how the processing technique with a cheap post-processor behaves in practice. Our purpose, rather than providing a complete analysis of different processed methods, is just to illustrate the previous theoretical analysis on some specific examples. We consider a kernel with effective order 6 of the form (2.14) with $s = 11$ constructed and studied in [18, 19]. Its coefficients $\alpha_i = \alpha_{12-i}$ are collected in Table 4.1. Next we construct an approximation $\pi_h^{(c)} \in \mathbb{P}_6$ to the post-processor π_h also as a composition of the 2nd-order integrator $\mathcal{S}_h$ at different stages. In particular, we take

$$\pi_h^{(c)} = \omega_h \circ \omega_{-h} \simeq \pi_h, \qquad \text{with} \qquad \omega_h = \mathcal{S}_{\gamma_6 h} \circ \cdots \circ \mathcal{S}_{\gamma_1 h} \tag{4.1}$$

and coefficients γ_i, $i = 1, \ldots, 6$ given in Table 4.1. Finally we consider the intermediate values (3.3) and solve equations (A.2) for the cheap post-processor $\hat{\pi}_h \in \mathbb{P}_5$. The corresponding solution obtained by taking w_1, w_5, w_6, w_7 (in addition to w_0) as the non-zero coefficients is also collected in Table 4.1.

TABLE 4.1
Coefficients for the 6th-order processed method with kernel ψ_h of the form (2.14) ($s = 11$) and post-processors π_h and $\hat{\pi}_h$ given by (4.1) and (3.8), respectively.

$\mathrm{P}_{11}6$		
$\alpha_1 = 0.1705768865009222157$	$\gamma_6 = -0.1$	$w_0 = 1 - 2(w_1 + w_5 + w_6 + w_7)$
$\alpha_2 = \alpha_1$	$\gamma_5 = 0.24687306977659$	$w_1 = 0.35601475536028$
$\alpha_3 = \alpha_1$	$\gamma_4 = 0.09086982276241$	$w_5 = 0.12246549694690$
$\alpha_4 = \alpha_1$	$\gamma_3 = 0.23651387483203$	$w_6 = 0.00415291514453$
$\alpha_5 = -0.4233661408926580 48$	$\gamma_2 = -0.20621953139126$	$w_7 = -0.20658995116781$
$\alpha_6 = 1 - 2(\alpha_1 + \cdots + \alpha_5)$	$\gamma_1 = -(\gamma_2 + \cdots + \gamma_6)$	

We recall that both $\pi_h^{(c)}$ and $\hat{\pi}_h$ are approximations to the post-processor π_h. The map $\pi_h^{(c)}$ is built as a composition of the basic integrator $\mathcal{S}_h$, so that $\log(\pi_h^{(c)}) \in \mathcal{L}$, and $\hat{\pi}_h$ is taken as a linear combination of intermediate values used in the calculation of the kernel. From (4.1) we observe that the computational cost of $\pi_h^{(c)}$ is similar to that of the kernel, whereas $\hat{\pi}$ can be considered cost free.

We compare this 6th-order test integrator with other standard non processed composition methods of the same family. In particular we consider the well known 6th-order seven stages method 'A' ($\mathrm{Y}_7 6$) built by Yoshida [25] and the optimized 6th-order nine stages method ($SS, m = 9$) of McLachlan [16] ($\mathrm{M}_9 6$) (similar results are obtained with the 6th-order nine stages method proposed by Kahan and Li [13]).

Numerical Example 1. To illustrate how the error is propagated along the evolution when different approximations to the post-processor are considered, we take the simple Lotka-Volterra problem

$$u' = u(v-2), \qquad v' = v(1-u), \tag{4.2}$$

which admits as first integral $I(u,v) = \ln(uv^2) - (u+v)$. Using logarithmic scale ($q = \ln v,\ p = \ln u$) the system becomes Hamiltonian with $H = p - e^p + 2q - e^q = T(p) + V(q)$. Equations (4.2) can be written as $x' = f_a(x) + f_b(x)$ with $x = (u,v)$, $f_a = (u(v-2), 0)$, $f_b = (0, v(1-u))$, so that the corresponding h-flows $\varphi_h^{[a]}$ and $\varphi_h^{[b]}$ can be exactly computed. We choose as 2nd-order time-symmetric integrator the composition $\mathcal{S}_h = \varphi_{h/2}^{[a]} \circ \varphi_h^{[b]} \circ \varphi_{h/2}^{[a]}$.

In the region $0 < u, v$ the system has periodic trajectories around $(u,v) = (1,2)$. We take $(u_0, v_0) = (1,1)$, integrate up to $t = 100 \times 2\pi$, and get outputs at $t = i \times 2\pi,\ i = 1, \ldots, 100$. In Figure 4.1(a) we present the global error for the processed schemes both using the accurate post-processor $\pi_h^{(c)}$ of (4.1) (method $\mathrm{P}_{11}6$) and the cheap approximation $\hat{\pi}_h$ ($\mathrm{P}_{11}6\mathrm{C}$) only for output. The results obtained are compared with $\mathrm{Y}_7 6$ and $\mathrm{M}_9 6$. The time steps selected are $h = \frac{1}{14}, \frac{1}{11}, \frac{1}{9}$, for $\mathrm{Y}_7 6$, $\mathrm{M}_9 6$ and $\mathrm{P}_{11}6$, respectively, so that all methods require approximately the same number of evaluations. Figure 4.1(b) shows the error in the first integral $I(u,v)$ for $\mathrm{P}_{11}6$ and $\mathrm{P}_{11}6\mathrm{C}$. In this case, for $1.9 < \log(t) \leq 2$ the cheap post-processor $\hat{\pi}_h$ is replaced by $\pi_h^{(c)}$ just to clearly show that this higher accuracy can always be recovered. If $\mathrm{P}_{11}6\mathrm{C}$ is started with $(\hat{\pi}_h)^{-1}$ instead of $\left(\pi_h^{(c)}\right)^{-1}$, this accuracy would not have been restored.

From the figures we observe: (a) the processed integrator is clearly more accurate; (b) the results for the global error obtained using $\hat{\pi}_h$ approach asymptotically those given by $\pi_h^{(c)}$; (c) the error in $I(u,v)$ is higher when $\hat{\pi}_h$ is used but it does not grow

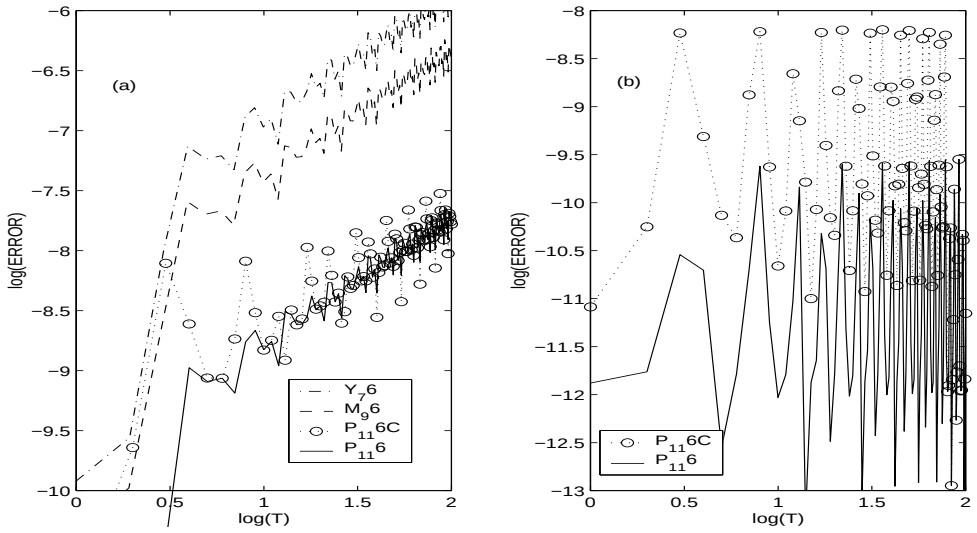

FIG. 4.1. *(a) Error in position and (b) error in the first integral $I(u,v)$ as functions of time for the Lotka-Volterra problem. The time step is chosen such that all methods require the same number of evaluations (this number corresponds to the kernel for the processed integrators).*

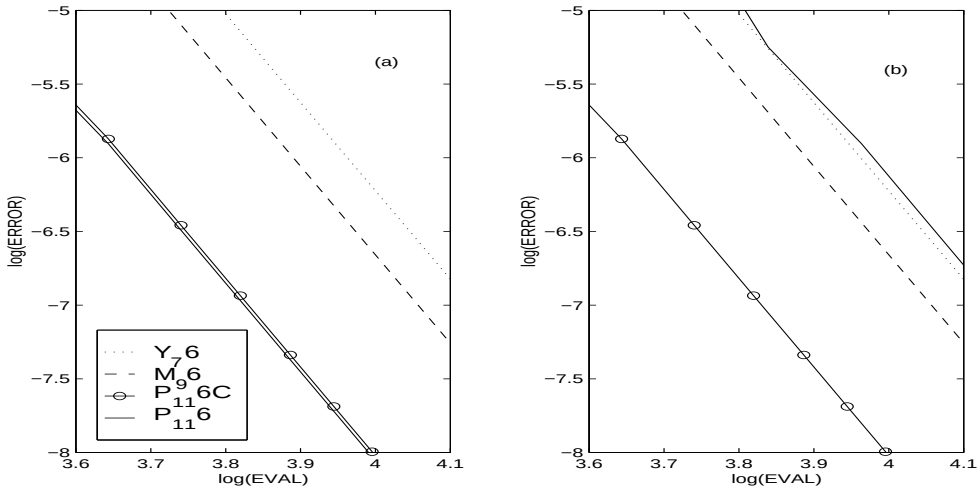

FIG. 4.2. *Average error in position versus number of evaluations for the first example (a) when the output is not frequent; (b) when the output is required at each step.*

with time, and the more accurate results can always be retrieved using $\pi_h^{(c)}$ when desired.

Next we measure the average relative error in position versus the number of evaluations for different time steps and methods. Figure 4.2 shows the results (a) when the output is required only occasionally and (b) when it is required at each step. From this figure the importance of using a cheap post-processor when the output is desired frequently is clear.

Numerical Example 2: Let us consider now the ABC-flow [12], whose equations

are given by

$$\begin{aligned} x' &= B\cos y + C\sin z \\ y' &= C\cos z + A\sin x \\ z' &= A\cos x + B\sin y \end{aligned} \tag{4.3}$$

and the vector field is separable in three solvable parts, i.e.,

$$f = f_a + f_b + f_c = A(0, \sin x, \cos x) + B(\cos y, 0, \sin y) + C(\sin z, \cos z, 0).$$

We take as initial condition $(x_0, y_0, z_0) = (3.14, 2.77, 0)$, parameters $A = B = C = 1$ and integrate the system until $t = 100$. We choose as the basic symmetric second order integrator $\mathcal{S}_h = \chi_{h/2} \circ \chi^*_{h/2}$, where $\chi_h = \varphi_h^{[a]} \circ \varphi_h^{[b]} \circ \varphi_h^{[c]}$ and $\chi^*_h = \varphi_h^{[c]} \circ \varphi_h^{[b]} \circ \varphi_h^{[a]}$. In Figure 4.3 we show the error growth in Euclidean norm when the following integrators based on $\mathrm{P}_{11}6$ are considered:

- ψ_h: only the kernel without pre- and post-processor (dash-dotted line, $\mathrm{K}_{11}4$);
- $\hat{\pi}_h \circ \psi_h \circ \hat{\pi}_h^{-1}$: the cheap pre- and post-processor are employed (dotted line, $\mathrm{P}_{11}6\mathrm{CC}$);
- $\hat{\pi}_h \circ \psi_h \circ (\pi_h^{(c)})^{-1}$: we use the accurate pre-processor and the cheap post-processor (circles joined by dotted lines, $\mathrm{P}_{11}6\mathrm{C}$);
- $\pi_h^{(c)} \circ \psi_h \circ (\pi_h^{(c)})^{-1}$: the accurate pre- and post-processor are used (solid line, $\mathrm{P}_{11}6$).

We also include the results obtained using $\mathrm{M}_9 6$ (dashed line), choosing the time step such that the number of evaluations is the same as for the kernel. From the figure, it is clear that the kernel by itself is not good enough for giving accurate results (it is only a fourth order integrator). In addition we see that, at least for this problem, it is important to start the computation using a good pre-processor (some accuracy is lost when using $\hat{\pi}_h^{-1}$). Finally, we observe that after some time the results obtained using the cheap and the composition post-processors agree up to drawing accuracy, but the former is faster to compute.

5. Concluding remarks. We have presented a general study of the processing technique which can be readily applied in several contexts. We obtain the number of order conditions and indicate how to find them explicitly in a systematic way. We have also presented a technique to find post-processors virtually cost free, just using intermediate results from the kernel. From the error propagation analysis we conclude that it is important to start the computation with an accurate pre-processor (even if it is expensive), and that, in general, a computationally cheap post-processor can be safely used for ordinary intermediate output, although a more expensive post-processor may be used, if required, to compute more accurate results at selected times.

An important application of the results contained in this paper is the construction of processed methods whose kernel is a composition of low-order basic integrators. In that case, by analyzing the structure of the corresponding Lie algebra $\mathcal{L}$, it is possible to obtain approximations to the post-processor either as a composition of basic methods or as a linear combination of intermediate stages of the kernel. In [2] this analysis is pursued in more detail for different families of composition methods, and new high order schemes are constructed which prove to be more efficient than other composition integrators available in the literature.

In practice, the efficient integration of systems of ODEs often requires the use of some step-size changing strategy. In principle, two possibilities can be contemplated.

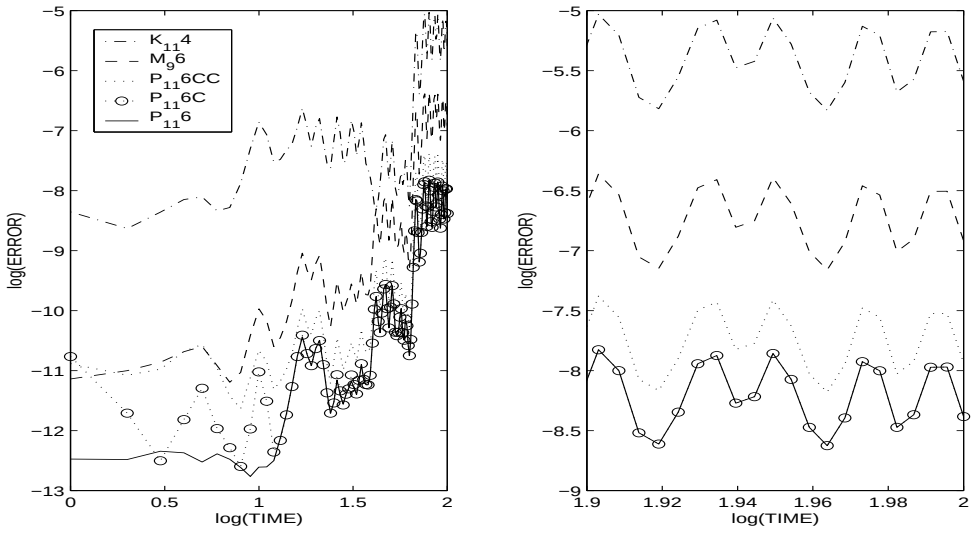

FIG. 4.3. *Error growth in position for the ABC-flow problem using a kernel (2.14) with coefficients in Table 4.1 and different pre- and post-processors. The results obtained with the non processed integrator* $\mathrm{M}_9 6$ *are also shown. The picture to the right is an enlargement of the rectangle* $[1.9, 2] \times [-9, -5]$ *in the left-hand picture.*

(i) Reparameterize the time variable in such a way that, with the new independent variable, a constant step-size can be used [12]. This is a familiar approach in geometric integration, and the theory developed here applies directly. (ii) Consider the problem of adapting the step-size in general terms, i.e., to construct processed methods whose step-size h changes to ρh, with $\rho \in [\rho_{\min}, \rho_{\max}]$ chosen according to some sort of local error estimation technique. This is the usual approach for general purpose integrators such as those based on explicit Runge–Kutta methods and it is not suitable for geometric integration, as such standard variable step-size implementation destroys the geometric nature of the integration [22]. Although recently an adaptation of processing techniques to standard variable step-size strategies has been proposed in the Runge–Kutta context [9], this is largely an open problem which deserves further research.

Appendix A.

Here we derive explicitly the effective order conditions up to order 6 for methods with kernel (2.14) and obtain the corresponding linear equations (3.9) for the cheap post-processor (3.2). The series $S_h = I + \sum_{k\geq 1} h^k S_k$ of differential operators associated with the second order time-symmetric integrator $\mathcal{S}_h : \mathbb{R}^D \longrightarrow \mathbb{R}^D$ for equation (1.1) can be written as $S_h = \exp(Y_h)$, where $Y_h = hY_1 + h^3 Y_3 + h^5 Y_5 + \cdots$, and $Y_1 = F$. Then

$$\Psi_h = \exp(Y_{h\alpha_1}) \cdots \exp(Y_{h\alpha_s}). \tag{A.1}$$

By repeated application of the Baker–Campbell–Hausdorff formula one arrives at an expansion of $F_h = \log(\Psi_h) = hF_1 + h^3 F_3 + h^4 F_4 + \cdots$, with $h^k F_k \in \mathcal{L}_k$ for the graded Lie algebra $\mathcal{L} = \bigoplus_{k\geq 1} \mathcal{L}_k$ generated by the vector fields $\{Y_1, Y_3, Y_5, \cdots\}$. Here $n_1 = 1$, $n_2 = 0$, $n_k^* = n_k$ for $k \geq 2$, whence, according to Lemma 2, $F_2 = 0$, $P_1 = P_2 = 0$. A basis of $\mathcal{L}$ is given in Table A.1 up to $k = 6$.

	Basis of $\mathcal{L}$	
$\mathcal{L}_1$	$E_{1,1} = Y_1 = F$	
$\mathcal{L}_3$	$E_{3,1} = Y_3$	
$\mathcal{L}_4$	$E_{4,1} = [F, E_{3,1}]$	
$\mathcal{L}_5$	$E_{5,1} = Y_5$	$E_{5,2} = [F, E_{4,1}]$
$\mathcal{L}_6$	$E_{6,1} = [F, E_{5,1}]$	$E_{6,2} = [F, E_{5,2}]$

TABLE A.1

Basis of $\mathcal{L} = \bigoplus_{k\geq 1} \mathcal{L}_k$, the free Lie algebra generated by $\{hY_1, h^3Y_3, h^5Y_5, \cdots\}$.

The order conditions for the kernel and post-processor up to order six in this basis read

$$f_{1,1} = 1, \qquad f_{3,1} = 0, \qquad f_{5,1} = 0$$
$$p_{4,1} = -f_{5,2}, \qquad p_{1,1} = p_{3,1} = p_{5,1} = p_{5,2} = 0.$$

The basis for $\mathcal{L}_k$ presented in Table A.1 leads to the following basis in the universal enveloping algebra:

$$\mathcal{A}_1 :\ D_{1,1} = E_{1,1}, \qquad \mathcal{A}_2 :\ D_{2,1} = \frac{1}{2}E_{1,1}^2,$$
$$\mathcal{A}_3 :\ D_{3,1} = E_{3,1}, \quad D_{3,2} = \frac{1}{3!}E_{1,1}^3,$$
$$\mathcal{A}_4 :\ D_{4,1} = E_{4,1}, \quad D_{4,2} = \frac{1}{4!}E_{1,1}^4, \quad D_{4,3} = \frac{1}{2}(E_{1,1}E_{3,1} + E_{3,1}E_{1,1}).$$

The series of vector fields Π_h corresponding to the optimal processor is

$$\Pi_h = \exp(P_h) = \exp\left(h^4 p_{4,1}E_{4,1} + \mathcal{O}(h^6)\right) = I + h^4 p_{4,1}D_{4,1} + \mathcal{O}(h^6).$$

For the intermediate stages of the cheap approximation we take (3.3), or equivalently

$$\Phi_h^{(i)} = \exp(Y_{h\alpha_1}) \cdots \exp(Y_{h\alpha_i}) = \exp\left(hf_{1,1}^{(i)}E_{1,1} + h^3 f_{3,1}^{(i)}E_{3,1} + h^4 f_{4,1}^{(i)}E_{4,1} + \mathcal{O}(h^5)\right).$$

Then $\Phi_h^{(i)} + \Phi_h^{(-i)} = 2\left(I + \Phi_2^{(i)}h^2 + \Phi_4^{(i)}h^4 + \mathcal{O}(h^6)\right)$, with

$$\Phi_2^{(i)} = \phi_{2,1}^{(i)}D_{2,1}, \qquad \Phi_4^{(i)} = \left(\phi_{4,1}^{(i)}D_{4,1} + \phi_{4,2}^{(i)}D_{4,2} + \phi_{4,3}^{(i)}D_{4,3}\right).$$

Here

$$\phi_{2,1}^{(i)} = (f_{1,1}^{(i)})^2, \qquad \phi_{4,1}^{(i)} = f_{4,1}^{(i)}, \qquad \phi_{4,2}^{(i)} = (f_{1,1}^{(i)})^4, \qquad \phi_{4,3}^{(i)} = f_{1,1}^{(i)}f_{3,1}^{(i)},$$

and

$$f_{1,1}^{(i)} = \sum_{j=1}^{i}\alpha_j; \qquad f_{3,1}^{(i)} = \sum_{j=1}^{i}\alpha_j^3; \qquad f_{4,1}^{(i)} = \frac{1}{2}\left(\sum_{j=1}^{i-1}\alpha_j\sum_{k=1}^{j}\alpha_k^3 - \sum_{j=1}^{i-1}\alpha_j^3\sum_{k=1}^{j}\alpha_k\right)$$

with $f_{4,1}^{(1)} = 0$. Finally, (3.9) for $k = 2, 4$ leads to the following linear system of equations:

$$\text{(A.2)}\quad \sum_{i=1}^{s}\phi_{2,1}^{(i)}w_i = 0; \quad \sum_{i=1}^{s}\phi_{4,1}^{(i)}w_i = \frac{1}{2}p_{4,1}; \quad \sum_{i=1}^{s}\phi_{4,2}^{(i)}w_i = 0; \quad \sum_{i=1}^{s}\phi_{4,3}^{(i)}w_i = 0,$$

so that $\hat{\pi}_h \in \mathbb{P}_5$.

REFERENCES

[1] S. Blanes, *High order numerical integrators for differential equations using composition and processing of low order methods*, Appl. Numer. Math., 37 (2001), pp. 289–306.

[2] S. Blanes, F. Casas, and A. Murua, *Composition methods for differential equations with processing*. In preparation.

[3] S. Blanes, F. Casas, and J. Ros, *Symplectic integrators with processing: a general study*, SIAM J. Sci. Comput., 21 (1999), pp. 711–727.

[4] S. Blanes, F. Casas, and J. Ros, *Processing symplectic methods for near-integrable Hamiltonian systems*, Celest. Mech. & Dyn. Astr., 77 (2000), pp. 17–35.

[5] S. Blanes, F. Casas, and J. Ros, *High-order Runge-Kutta-Nyström geometric integrators with processing*, Appl. Numer. Math., 39 (2001), pp. 245–259.

[6] J. Butcher, *The Numerical Analysis of Ordinary Differential Equations*, John Wiley & Sons Ltd,. 1987.

[7] J. Butcher, *The effective order of Runge-Kutta methods*. In: Conference on the numerical solution of differential equations, Lecture Notes in Math., Vol. 109 Springer, Berlin, (1969), pp. 133–139.

[8] J. Butcher and J.M. Sanz-Serna, *The number of conditions for a Runge-Kutta method to have effective order p*, Appl. Numer. Math., 22 (1996), pp. 103–111.

[9] J.C. Butcher and T.M.H. Chan, *Variable stepsize schemes for effective order methods and enhanced order composition methods*, Numer. Algorithms, 26 (2001), pp. 131-150.

[10] S.K. Gray and D.E. Manolopoulos, *Symplectic integrators tailored to the time-dependent Schrödinger equation*, J. Chem. Phys., 104 (1996), pp. 7099-7112.

[11] E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I*, 2nd Ed., Springer, Berlin, 1993.

[12] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer, Berlin, 2002.

[13] W. Kahan and R.C. Li, *Composition constants for raising the order of unconventional schemes for ordinary differential equations*, Math. Comp., 66 (1997), pp. 1089–1099.

[14] M.A. López-Marcos, J.M. Sanz-Serna, and R.D. Skeel, *Cheap enhancement of symplectic integrators*, in Proceedings, 1995 Dundee Conference on Numerical Analysis, D.F. Griffiths, G.A. Watson, Eds. Longman Group (1996).

[15] M.A. López-Marcos, J.M. Sanz-Serna, and R.D. Skeel, *Explicit symplectic integrators using Hessian-vector products*, SIAM J. Sci. Comput., 18 (1997), pp. 223–238.

[16] R.I. McLachlan, *On the numerical integration of ordinary differential equations by symmetric composition methods*, SIAM J. Sci. Comput., 16 (1995), pp. 151–168.

[17] R.I. McLachlan, *More on symplectic correctors*, in Integration Algorithms and Classical Mechanics, Vol. 10, J.E. Marsden, G.W. Patrick, and W.F. Shadwick, eds., American Mathematical Society, Providence, R.I., (1996), pp. 141–149.

[18] R.I. McLachlan, *Families of high order composition methods*, Numer. Algorithms, 31 (2002), pp. 233–246.

[19] R.I. McLachlan and G.R.W. Quispel, *Splitting methods*, Acta Numerica, 11 (2002), 341–434.

[20] H. Munthe-Kaas and B. Owren, *Computations in a free Lie algebra*, Philosophical Trans. Royal Soc. A, 357 (1999), pp. 957–981.

[21] P.J. Olver, *Applications of Lie Groups to Differential Equations*, 2nd. Ed., Springer, New York, 1993.

[22] J.M. Sanz-Serna and M.P. Calvo, *Numerical Hamiltonian Problems*, Chapman & Hall, London, 1994.

[23] V.S. Varadarajan, *Lie Groups, Lie Algebras and their Representations*, Springer, New York, 1984.

[24] J. Wisdom, M. Holman, and J. Touma, *Symplectic correctors*, in Integration Algorithms and Classical Mechanics, Vol. 10, J.E. Marsden, G.W. Patrick, and W.F. Shadwick, eds., American Mathematical Society, Providence, R.I., (1996), pp. 217–244.

[25] H. Yoshida, *Construction of higher order symplectic integrators*, Phys. Lett. A, 150 (1990), pp. 262–268.