RUNGE–KUTTA LAWSON SCHEMES FOR STOCHASTIC DIFFERENTIAL EQUATIONS

KRISTIAN DEBRABANT AND ANNE KVÆRNØ AND NICKY CORDUA MATTSSON

ABSTRACT. In this paper, we present a framework to construct general stochastic Runge–Kutta Lawson schemes. We prove that the schemes inherit the consistency and convergence properties of the underlying Runge–Kutta scheme, and confirm this in some numerical experiments. We also investigate the stability properties of the methods and show for some examples, that the new schemes have improved stability properties compared to the underlying schemes.

1. Introduction

Stochastic differential equations (SDEs) are an essential tool in order to model and understand real-life systems under the influence of noise, see, e.g., [18, Section 7.1–7.10] for examples. We do, however, only know the exact solution to very few of these equations, thus will usually have to integrate the differential equations numerically. To efficiently do this, it is of interest to construct numerical schemes that can recreate essential dynamics of the exact solution.

Linear terms in the drift and the diffusion can often represent these essential dynamics. See e.g. recent works on linear stability analysis [5, 4, 6] and highly oscillatory differential equations [11, 13]. For the same reason, much work has also gone into treating these parts explicitly, see e.g. the work on local linearization techniques by [3, 17, 10]. In this paper we assume that the relevant linear terms have been made explicitly available, and thus consider SDEs of the form

\[ dX(t) = \sum_{m=0}^{M} (A_mX(t) + g_m(t, X(t))) \star dW_m(t), \quad X(t_0) = X_0, \]

where \( W_m \) for \( m = 1, \ldots, M \) denote independent scalar Wiener processes, \( W_0(t) = t \) denotes the time and the SDE is solved on the interval \( I = [t_0, T] \). Here, the stochastic integral can be interpreted as Itô integral with \( \star dW_m = dW_m \), or as Stratonovich integral with \( \star dW_m = dW_m \). We assume that SDE (1) has a unique solution for \( X_0 \in \mathbb{R}^d \) and that all \( g_m \) have the appropriate regularity for this (depends on the interpretation of the integral). We also assume that the matrices \( A_m \in \mathbb{R}^{d \times d}, m = 0, \ldots, M \), are constant and are chosen in connection with \( g_m \) such that the following assumption holds:

Assumption 1 (Commutativity).

\[ [A_l, A_k] = A_l A_k - A_k A_l = 0 \quad \text{for all} \quad l, k = 0, 1, \ldots, M. \]

Exponential integrators have, especially in the deterministic case, been very efficient at solving some types of differential equations. In the more recent years, much work has gone into extending these results and schemes to SDEs. In particular, Erdogan and Lord [15] construct an exponential Euler and an exponential Milstein scheme, including both the linear drift and diffusion, and numerically demonstrate that these schemes are more efficient.
on specific problems than their underlying schemes. They also show that for linear diffusion, the strong order of convergence of the exponential Euler scheme is $p = 1$. We also mention the work by [19,20], who apply specific exponential schemes to a stiff system and construct an explicit weak second-order exponential scheme that proves to be A-stable for the linear test-equation suggested by [16].

In this paper, we construct exponential integrators including both the linear drift and diffusion using the entire class of stochastic Runge-Kutta (SRK) schemes and a stochastic extension of Lawson type schemes (also known as integrating factor methods) for both Stratonovich and Itô integrals. We prove that these stochastic Lawson (SL) schemes, under some conditions, inherit both the strong and weak order of convergence of the underlying scheme, and provide a general framework to construct higher-order exponential schemes.

In Section 2 we present a stochastic extension of the deterministic Lawson transformation [22] and derive then the general class of SRK Lawson schemes, providing also several examples. In Section 3, we prove that these methods, under some conditions, converge both strongly and weakly of the same order as the underlying SRK schemes. These results are accompanied by numerical examples. In Section 4 we provide some linear stability analyses for a selection of these schemes, showing that exponential SRK schemes may have improved stability properties. We also compare these methods to the drift implicit Platen scheme, where we show that the Platen Lawson scheme better catches the behaviour of the reference solution for the considered examples. These results are also verified by numerical simulations.

2. Construction of SRK Lawson schemes

In this section, we present the overall idea of Lawson type schemes; we then apply the idea to the class of SRK schemes and provide several examples.

2.1. General construction. This section aims at constructing a numerical scheme which solves linear SDEs of the form

\begin{equation}
\frac{dX(t)}{dt} = \sum_{m=0}^{M} A_m X(t) \star dW_m(t), \quad X(t_0) = X_0
\end{equation}

exactly. Under Assumption [1], the exact solution of (2) can be written as [2,15]

\begin{equation}
X(t) = \exp \left[ \left( A_0 - \gamma^* \sum_{m=1}^{M} A_m^2 \right) (t-t_0) + \sum_{m=1}^{M} A_m (W_m(t) - W_m(t_0)) \right] X_0,
\end{equation}

with $\gamma^* = \frac{1}{2}$ in the Itô case, $\gamma^* = 0$ in the Stratonovich case (see also Lemma [1]). This exact solution will be used to construct the exponential integrators, in accordance to the approach to construct deterministic Lawson schemes [22].

Before we do so, we want to emphasize that Assumption [1] is not a restriction on the SDE to be considered, but rather a restriction on how to pick the matrices $A_m$. For demonstration consider the SDE

\begin{equation}
\frac{dX(t)}{dt} = (\tilde{A}_0 X(t) + f_0(t, X(t))) \, dt + (\tilde{A}_1 X(t) + f_1(t, X(t))) \star dW(t),
\end{equation}

where $\tilde{A}_0$ does not commute with $\tilde{A}_1$. It is always possible to find a splitting of $\tilde{A}_0$, $\tilde{A}_0 = A_0 + A'_0$, such that $A_0$ commutes with $\tilde{A}_1$, and we can thus rewrite (4) in the form of (1) by choosing $g_0(t, X(t)) = f_0(t, X(t)) + A'_0 X(t)$, $g_1(t, X(t)) = f_1(t, X(t))$, $A_1 = \tilde{A}_1$. 

An obvious choice is $A_0 = c\bar{A}$ for some scalar $c$. Optimally $A_0$ should represent the properties of the SDE that we are interested in simulating exactly. Conversely, we can also put $A_1 = c_1\bar{A}_0$, which ensures that $A_1$ commutes with $\bar{A}_0$.

Let a discretization $I^n = \{t_0, t_1, \ldots, t_N\}$ with $t_0 < t_1 \ldots < t_N = T$ of the time interval $I$ be given with $h_n = t_{n+1} - t_n$ for $n = 0, 1, \ldots, N - 1$ denoting the step size. To construct numerical schemes that solve (2) exactly, we will use the following lemma.

**Lemma 1.** Let $X$ be the solution of SDE (1) and let Assumption 1 be true. Then the locally transformed variable $V^n$ defined by

\[ V^n(t) = e^{-L^n(t)} X(t) \]

with

\[ L^n(t) = \left(A_0 - \gamma^* \sum_{m=1}^M A_m^2 \right) (t - t_n) + \sum_{m=1}^M A_m (W_m(t) - W_m(t_n)) \]

satisfies the SDE

\[ dV^n(t) = \sum_{m=0}^M \tilde{g}_m^n (W(t), V^n(t)) \circ dW_m(t), \quad V^n(t_n) = X(t_n), \]

where $W(t) = (W_m(t))_{m=0}^M$ and

\[ \tilde{g}_m^n(W(t), x) := e^{-L^n(t)} \bar{g}_m(t, e^{L^n(t)} x) \]

with

\[ \bar{g}_m(t, x) := \begin{cases} g_0(t, x) - 2\gamma^* \sum_{m=1}^M A_m g_m(t, x), & m = 0, \\ g_m(t, x), & m > 0. \end{cases} \]

**Proof.** Using Assumption 1 when applying Itô’s lemma to the transformed variable $V^n$ defined in (5) yields

\[
dV^n(t) = -\left(A_0 - \gamma^* \sum_{m=1}^M A_m^2 \right) e^{-L^n(t)} X(t) dt - \sum_{m=1}^M A_m e^{-L^n(t)} X(t) dW_m \\
+ \gamma^* \sum_{m=1}^M A_m^2 e^{-L^n(t)} X(t) dt + e^{-L^n(t)} dX(t) \\
+ 2\gamma^* \sum_{m=1}^M \left(-A_m e^{-L^n(t)} \right) \left(A_m X(t) + g_m(t, X(t)) \right) dt.
\]

Inserting (4) we obtain (again using Assumption 1)

\[
dV^n(t) = -2\gamma^* e^{-L^n(t)} \sum_{m=1}^M A_m g_m(t, X(t)) dt + e^{-L^n(t)} \sum_{m=1}^M g_m(t, X(t)) \circ dW_m(t)
\]

which together with (5) yields the assertion. $\square$

Note that by Lemma 1 every solution of (1) induces a solution of (7). Analogously one can also show that every solution $V^n$ of (7) induces a solution $X(t) = e^{L^n(t)} V^n(t)$ of (1).

As (1) is assumed to have a unique solution, this holds thus also for (7).

Note further that for linear SDEs (2), the right hand side of (7) will vanish. Denoting by $Y_n$ the discrete-time approximation to $X(t_n)$, we now define the family of one-step SL schemes by Algorithm 1.
Algorithm 1 Stochastic Lawson scheme

(a) Start with \( Y_0 = x_0 \).

(b) For \( n = 0, \ldots, N - 1 \)

(i) Apply one step of a one-step method to get an approximation \( V_{n+1}^n \) of the exact solution \( V^n(t_{n+1}) \) of

\[
dV^n(t) = \sum_{m=0}^{M} \tilde{g}^n_m(W(t), V^n(t)) \ast dW_m(t), \quad V^n(t_n) = Y_n
\]

at \( t_{n+1} \).

(ii) Define \( Y_{n+1} = e^{\Delta L^n} V_{n+1}^n \)

where \( \Delta L^n = \left( A_0 - \gamma^{*} \sum_{m=1}^{M} \hat{A}^2_m \right) (t_{n+1} - t_n) + \sum_{m=1}^{M} A_m \Delta W^m_n \in \mathbb{R}^{d \times d} \) with \( \Delta W^m_n \) being the approximation of \( W_m(t_{n+1}) - W_m(t_n) \) used by the one-step method applied in step (i).

We note that if \( L^n(t) \) is a stochastic process or if we use variable step-sizes, \( e^{L^n(t)} \) has to be recalculated at every step, which might be expensive for large dimensions \( d \). One way to avoid doing this is to pick \( A_m = 0 \) for \( m > 0 \) and to use constant step sizes, \( h_n = h \); then one only has to calculate the exponentials once. We denote such schemes as drift SL (DSL) schemes. In contrast, a full stochastic Lawson (FSL) scheme is a scheme with at least one nonzero linear diffusion term present in the operator \( L^n(t) \). The underlying numerical scheme is restored by setting all the linear parts to 0. This is typically the case if the linear part is a result of a spatial discretization of a diffusion term. Compared to the original schemes the computational overhead of the DSL schemes is negligible, as the matrix exponential is deterministic and is calculated only once. For the FSL schemes, this is not the case. Thus the applicability of the FSL methods depends on how efficient the matrix exponentials can be calculated, weighted with the advantages of improved performance. See [11][13] for some successful examples.

We now apply this general idea to the class of SRK schemes.

2.2. SRK Lawson schemes. To apply an SRK method to (7) we first have to transform (7) to an autonomous system, i.e. add \( M + 1 \) equations in order to obtain an autonomous SDE in terms of \( \bar{V}^n(t) = (W(t)^\top, V^n(t)^\top)^\top \).

\[
d\bar{V}^n(t) = \sum_{m=0}^{M} \tilde{g}^n_m(\bar{V}^n(t)) \ast dW_m(t), \quad \bar{g}^n_m = (\delta_{0,m}, \ldots, \delta_{M,m}, \tilde{g}^n_m)^\top, \quad \delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}
\]

Using the same notation as in [9], an \( s \)-stage SRK method applied to (9) is given by

\[
\bar{H}_i = \bar{V}_n + \sum_{j=1}^{s} \sum_{m=0}^{M} \bar{I}^{m,n}_{ij} \bar{g}^n_m(\bar{H}_j),
\]

\[
\bar{V}_{n+1}^n = \bar{V}_n + \sum_{i=1}^{s} \sum_{m=0}^{M} \bar{c}^{m,n}_{i} \bar{g}^n_m(\bar{H}_i)
\]

with suitable random variables \( \bar{Z}^{m,n}_{ij} \) and \( \bar{z}^{m,n}_{i} \). Letting

\[
c^{n,i}_m = \sum_{j=1}^{s} \bar{Z}^{m,n}_{ij}, \quad c^{n}_m = \sum_{i=1}^{s} \bar{z}^{m,n}_{i}
\]
it follows for the approximations $V_{n+1}^{m}$ to $V^{n}(t_{n+1})$ and $W^{n}$ to $W(t_{n})$ and the corresponding stage values that

$$H_{i} = V_{n}^{m} + \sum_{j=1}^{s} \sum_{m=0}^{M} Z_{ij}^{m,n} \hat{g}_{m}(W^{n} + c_{m}^{n,i}, H_{j}),$$

$$W_{m+1}^{n} = W_{m}^{n} + c_{m}^{n}, \quad m = 0, \ldots, M,$$  

$$V_{n+1}^{m} = V_{n}^{m} + \sum_{i=1}^{s} \sum_{m=0}^{M} c_{i}^{m,n} \hat{g}_{m}(W^{n} + c_{m}^{n,i}, H_{i}),$$

where $W^{n} + c_{m}^{n,i} = (W_{m}^{n} + c_{m}^{n,i})_{m=0}^{M}$. Defining the discrete updates $\Delta W_{m}^{n} = W_{m+1}^{n} - W_{m}^{n} = c_{m}^{n},$

$$\Delta L_{i}^{n} = \left(A_{0} - \gamma^{*} \sum_{m=1}^{M} A_{m}^{2}\right) c_{0}^{n,i} + \sum_{m=1}^{M} A_{m} c_{m}^{n,i},$$

and

$$\Delta L^{n} = \left(A_{0} - \gamma^{*} \sum_{m=1}^{M} A_{m}^{2}\right) \Delta W_{0}^{n} + \sum_{m=1}^{M} A_{m} \Delta W_{m}^{n}$$

and using the particular form of $\hat{g}$ and transforming back, we get the family of SRK Lawson schemes

$$H_{i} = Y_{n}^{m} + \sum_{j=1}^{s} \sum_{m=0}^{M} Z_{ij}^{m,n} e^{-\Delta L_{i}^{n}} \tilde{g}_{m}(t_{n} + c_{m}^{n,i}, e^{\Delta L_{i}^{n}} H_{j}),$$

$$V_{n+1}^{m} = Y_{n}^{m} + \sum_{i=1}^{s} \sum_{m=0}^{M} c_{i}^{m,n} e^{-\Delta L_{i}^{n}} \tilde{g}_{m}(t_{n} + c_{m}^{n,i}, e^{\Delta L_{i}^{n}} H_{i}),$$

$$Y_{n+1}^{m} = e^{\Delta L^{n}} Y_{n}^{m+1}.$$  

We will now look at some specific examples.

2.3. Some examples of SRK Lawson schemes. In the following, we will shortly discuss two SRK Lawson schemes for Itô SDEs, the Euler–Maruyama SL scheme and the Platen SL scheme, as well as the midpoint SL scheme for Stratonovich SDEs. In all cases, we assume that $\Delta W_{0}^{n} = \tilde{h}_{n}$ and that for $m \geq 1$, $\Delta W_{m}^{n}$ is a suitable approximation to $W_{m}(t_{n+1}) - W(t_{n})$, i.e., when we are interested in mean-square convergence of order $p$, then it needs to hold that $E(\Delta W_{m}^{n}) = O(h_{n+1}^{p})$ and $E[(\Delta W_{m}^{n})^{2}] = O(h_{n+1}^{2p+1})$, while only the first 3 moments of $\Delta W_{m}^{n}$ need to coincide with the ones of $W_{m}(t_{n+1}) - W(t_{n})$ if we are interested in weak convergence of order 1.

2.3.1. Euler–Maruyama SL scheme. The Euler–Maruyama scheme has the SRK coefficients $s = 1, Z_{11}^{m,n} = 0$, and $c_{1}^{m,n} = \Delta W_{m}^{n}$ and is mean square convergent of order 0.5 and weakly convergent of order 1. Following (13) the corresponding Euler–Maruyama SL scheme is given by

$$Y_{n+1}^{m} = e^{\Delta L^{n}} Y_{n}^{m} + e^{\Delta L^{n}} \sum_{m=0}^{M} \tilde{g}_{m}(t_{n}, Y_{n}) \Delta W_{m}^{n}.$$  

This scheme is, depending on how much of the linear diffusion is included in the exponential operator, also known as E0, HomE0 or Lawson Euler scheme, see e.g. [15][19].
2.3.2. Platen SL scheme. The Platen scheme for \( M = 1 \) \cite{Platen1992} Chapter 11.1] has the SRK coefficients \( s = 2, \ z_{1,1}^{m,n} = h_n, \ z_{2,1}^{m,n} = \Delta W_n \) \( = \frac{1}{2N}((\Delta W_n)^2 - h_n) \), \( z_{2,2}^{m,n} = 0, \ z_{2,1}^{1,n} = h_n, \ z_{2,1}^{1,n} = \sqrt{h_n}, \ z_{2,2}^{m,n} = 0 \) and is mean square convergent of order 1. According to \cite{Debrabant2010} the resulting Platen SL scheme is given by

\[
H_2 = Y_n + \bar{g}_0(t_n, Y_n) h_n + \bar{g}_1(t_n, Y_n) \sqrt{h_n},
\]

\[
V_n = Y_n + \tilde{g}_0(t_n, Y_n) h_n + \tilde{g}_1(t_n, Y_n) \Delta W_n + \frac{(\Delta W_n)^2 - h_n}{2 \sqrt{h_n}} \times \left[ e^{-(A_0 - \gamma^* A_1^*) h_n - A_1^* \sqrt{h_n}} \tilde{g}_1(t_n + h_n, e^{(A_0 - \gamma^* A_1^*) h_n + A_1^* \sqrt{h_n}} H_2) - \tilde{g}_1(t_n, Y_n) \right],
\]

\[
Y_{n+1} = e^{\Delta L_n^m} V_{n+1}.
\]

2.3.3. Midpoint SL scheme. The SRK coefficients for the stochastic implicit midpoint rule \cite{Lawson1977} (2.39) are given by \( s = 1, \ z_{1,1}^{m,n} = 1/2 \Delta W_n, \ z_{1,1}^{m,n} = \Delta W_n \). For commutative noise, the midpoint rule is mean square convergent of order 1, otherwise of order 0.5. Applying \cite{Debrabant2010} we obtain

\[
H_1 = Y_n + \sum_{m=0}^{M} \frac{1}{2} e^{-1/2 \Delta L_n} \tilde{g}_m \left( t_n + \frac{h_n}{2}, e^{1/2 \Delta L_n} H_1 \right) \Delta W_m,
\]

\[
V_n = Y_n + \sum_{m=0}^{M} e^{-1/2 \Delta L_n} \tilde{g}_m \left( t_n + \frac{h_n}{2}, e^{1/2 \Delta L_n} H_1 \right) \Delta W_m,
\]

\[
Y_{n+1} = e^{\Delta L_n^m} V_{n+1},
\]

which, by using that \( H_1 = \frac{1}{2} (V_{n+1} + Y_n) \), can be rewritten as

\[
Y_{n+1} = e^{\Delta L_n^m} Y_n + \sum_{m=0}^{M} e^{1/2 \Delta L_n} \tilde{g}_m \left( t_n + \frac{h_n}{2}, e^{1/2 \Delta L_n^m} Y_n + e^{-1/2 \Delta L_n^m} Y_{n+1} \right) \Delta W_m.
\]

3. Convergence of SRK Lawson schemes

In this section, we will prove that the class of SRK Lawson schemes that we just constructed has, under some conditions, the same order of convergence as the underlying SRK scheme, thus removing the need for individual convergence proofs for the individual methods. Afterwards we will give some numerical examples. To simplify the presentation, from now on we will restrict to equidistant step sizes \( h_n = h = \frac{T - t_0}{N}, n = 0, \ldots, N - 1 \).

3.1. Strong and weak convergence. To prove that the SRK Lawson schemes, under some conditions, inherit the consistency and convergence of the underlying SRK scheme, we first introduce global Lawson schemes in Algorithm \cite{Debrabant2010} Even though Algorithm \cite{Debrabant2010} looks similar to Algorithm \cite{Debrabant2010} there are significant differences: In Algorithm \cite{Debrabant2010} we integrate from \( t_{n-1} \) to \( t_n \), transform back and then define a new SDE for \( V_n \). In contrast, in Algorithm \cite{Debrabant2010} we integrate a single SDE for \( V^0(t) \) from \( t_0 \) to \( t_n \) before transforming back.

For Algorithm \cite{Debrabant2010} we can prove that the global Lawson scheme inherits the strong convergence of the underlying one-step method:

**Lemma 2** (Strong convergence of Algorithm \cite{Debrabant2010}). Let Assumption \cite{Debrabant2010} hold and let \( V_n^0 \) be the numerical approximation of \( \int_{t_n}^{t} V^0(s) ds \) by some one-step method of mean square order \( p \), i.e., there exists a \( c \in \mathbb{R} \) such that for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \) it holds that

\[
\sqrt{E(||V_n^0 - V^0(t_n)||^2)} \leq cn^0. \]

Moreover, assume that \( L_n^0 = L^0(t_n), i.e., \( W_n = W(t_n), \)
Algorithm 2 Global stochastic Lawson scheme

(a) Apply a one-step method to get approximations \( V_{n+1}^0 \) of the exact solution \( V^0(t_{n+1}) \) of

\[
dV^0(t) = \sum_{m=0}^{M} S_m(W(t), V^0(t)) \, dW_m(t), \quad V^0(t_0) = x_0
\]

for \( n = 0, \ldots, N - 1 \).

(b) For \( n = 0, \ldots, N \), define

\[
Y_n = e^{L_n^0} V_n^0
\]

where \( L_n^0 = \left( A_0 - \gamma \sum_{m=1}^{M} A_m^2 \right) (t_n - t_0) + \sum_{m=1}^{M} A_m (W_m^0 - W_m^0) \in \mathbb{R}^{d \times d} \) with \( W_m^0 \) being the approximation of \( W_m(t_n) \) induced by the one-step method used in step (i).

and that \( X \) is solution of SDE \( \{1\} \). Then there exists a \( \bar{c} \in \mathbb{R} \) such that the numerical approximation \( Y_n = e^{L_n^0} V_n^0 \) satisfies for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \)

\[
E\|Y_n - X(t_n)\|_2 \leq \bar{c} h^p.
\]

Here, \( \| \cdot \|_2 \) denotes the Euclidean vector norm.

**Proof.** The strong convergence of Algorithm 2 follows from \( e^{L_n^0} \) being mean square bounded \( \{13\} \) Lemma 5.1, thus

\[
E\|Y_n - X(t_n)\|_2 = E\|e^{L_n^0} V_n^0 - e^{L_n^0} V^0(t_n)\|_2 \\
\leq c_1 \sqrt{E\|V_n^0 - V^0(t_n)\|_2^2} \\
\leq c_1 h^p,
\]

where we also used \( \{18\}, \{5\} \) (with \( n = 0 \)), and the Cauchy–Schwarz inequality. \( \square \)

A function \( g : \mathbb{R}^n \to \mathbb{R} \) is called polynomially bounded if there exist constants \( K \) and \( \kappa \geq 0 \) such that for all \( x \in \mathbb{R}^n \) it holds that

\[
|g(x)| \leq K (1 + \|x\|_\kappa^\kappa).
\]

In the following we denote by \( C^2_p(\bar{\rho} + 1)(\mathbb{R}^n, \mathbb{R}) \) the class of functions \( g \in C^2(\bar{\rho} + 1)(\mathbb{R}^n, \mathbb{R}) \) for which \( g \) and all its partial derivatives of order up to \( 2(\bar{\rho} + 1) \), inclusively, are polynomially bounded. With this definition in place, we can also prove that the global Lawson scheme inherits the weak order of convergence of the underlying one-step method:

**Lemma 3** (Weak convergence of Algorithm 2). Let Assumption \( \{7\} \) hold and assume that \( X \) is solution of SDE \( \{1\} \). Moreover, assume that

(a) \( A_m \) for \( m > 0 \) are skew-symmetric,

(b) \( W^n \) and \( V_n^0 \) are the numerical approximations of \( W(t_n) \) and \( \{17\} \) obtained by applying some one-step method to the autonomous system \( \{9\} \), and we assume especially that \( W_0^0 = W_0(t_n) = t_n \).

(c) this approximation is of weak order \( \bar{\rho} \), i.e. for all \( g \in C^2_p(\bar{\rho} + 1)(\mathbb{R}^{M+1} \times \mathbb{R}^d, \mathbb{R}) \) there exists a \( c \in \mathbb{R} \) such that for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \) it holds that \( |E(g(W^n, V_n^0) - g(W(t_n), V^0(t_n)))| \leq c h^p \).
Then there exists for all \( f \in C_P^{2(\hat{p}+1)}(\mathbb{R}^d, \mathbb{R}) \) a \( \tilde{c} \in \mathbb{R} \) such that the numerical approximation \( Y_n = e^{t_n}V_n^0 \) satisfies for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \)
\[
|f(Y_n) - f(X(t_n))| \leq \tilde{c}h^\hat{p}.
\]

**Proof.** For \( s \in I = [t_0, T] \) and \( x \in \mathbb{R}^M, y \in \mathbb{R}^d \) let
\[
\varphi_s(x, y) = f(e^{(A_0+y^* \Sigma M_{m=1} A_m)(s-t_0)} e^{\sum_{m=1} M A_m(x_m-W_m^0)} y).
\]
Then it follows by (b) that
\[
f(Y_n) = \varphi_{t_n}(W_n, V_n^0), \quad f(X(t_n)) = \varphi_{t_n}(W(t_n), V_n^0(t_n)).
\]
To prove the assertion it is therefore by (c) enough to show that \( \varphi_s \) and all its derivatives up to order \( 2(\hat{p}+1) \) satisfy an estimate of the form
\[
||\varphi_s(x, y)||_2 \leq K(1 + ||x||_2^{2\hat{p}} + ||y||_2^{2\hat{p}})
\]
uniformly for all \( s \in I \) (see [23 Chapter 8]), which follows from \( f \in C_P^{2(\hat{p}+1)} \) and \( ||e^{\sum_{m=1} M A_m(x_m-W_m^0)}||_2 = 1 \) due to (a) and the exponential of a skew symmetric matrix having Euclidean norm 1. \( \square \)

Using a DSL scheme trivially satisfies (a) and Assumption 1 and thus any DSL scheme immediately inherits the weak convergence of the underlying one-step method.

We next demonstrate that for a given underlying SRK scheme, the local and the global Lawson scheme (Algorithms 1 and 2) result in the same output. Based on the autonomous form [9], an SRK method applied to the autonomous version of the globally transformed equation (17) can be written as
\[
H_i^0 = V_i^0 + \sum_{j=1}^s \sum_{m=0}^M \left| Z_{ij}^m e^{-t_i^0} \tilde{g}_m(t_n+c_0^{m,j}, e^{-t_i^0} H_i^0),
\]
(21)
\[
V_{n+1}^0 = V_n^0 + \sum_{i=1}^s \sum_{m=0}^M \left| \tilde{Z}_i^m e^{-t_i^0} \tilde{g}_m(t_n+c_0^{m,i} e^{-t_i^0} H_i^0)
\]
with
\[
L_i^0 = \left( A_0 - y^* \sum_{m=1}^M A_m \right)(t_n+c_0^{m,i} - t_0) + \sum_{m=1}^M A_m(W_m^0 + c_0^{m,i} - W_m^0)
\]
and with \( \tilde{g}_m(t, x) \) defined in Lemma 1. Using this formulation, we get the following result:

**Lemma 4.** Algorithm 1 with an arbitrary SRK method [10] applied to [8] and Algorithm 2 with the same underlying SRK method [21] applied to [17] give the same sequence of approximation points \( Y_n \).

**Proof.** We prove this by induction. Let \( Y_n \) denote the numerical solution by Algorithm 1. Similarly, let \( e^{t_n}V_n^0 \) be the numerical approximation obtained by Algorithm 2. It holds that \( Y_0 = e^{t_0}V_0^0 = x_0 \). We assume now that it holds that \( e^{t_n}V_n^0 = Y_n \), and prove that this implies that \( e^{t_{n+1}}V_{n+1}^0 = Y_{n+1} \).

For this, we consider a single update from \( t_n \) to \( t_{n+1} \) using the SRK method (21) for Algorithm 2. Defining \( H_i = e^{t_i}H_i^0 \) and multiplying \( V_{n+1}^0 \) by \( e^{t_{n+1}} \) it holds then
\[
H_i = e^{t_i}V_n^0 + \sum_{j=1}^s \sum_{m=0}^M \left| Z_{ij}^m e^{-t_i} \tilde{g}_m(t_n+c_0^{m,j} e^{-t_i} L_i^0) H_j^0)
\]
Then there exists a \( \tilde{W}^n \) which is identical to Algorithm 1 given in (13), and consequently it holds that

\[
Y_{n+1} = e^{\tilde{L}^0_n} V^0_n + \sum_{i=1}^{s} \sum_{m=0}^{M} c^{m,n}_i e^{L^0_n - L^0_m} g_m(t_n + c^{n,i}_0, e^{\Delta L^n} H_i).
\]

Using the induction hypothesis, \( Y_n = e^{\tilde{L}^0_n} V^0_n \), and \( L^0_n - L^0_m = \Delta L^n \), we obtain

\[
H_i = Y_n + \sum_{j=1}^{s} \sum_{m=0}^{M} Z^{m,n}_{i,j} e^{-\Delta L^n} \tilde{g}_m(t_n + c^{n,j}_0, e^{\Delta L^n} H_j),
\]

\[
e^{\tilde{L}^0_n} V^0_n = e^{\tilde{L}^0_n - L^0_n} Y_n + \sum_{i=1}^{s} \sum_{m=0}^{M} c^{m,n}_i e^{L^0_n - L^0_m} \tilde{g}_m(t_n + c^{n,i}_0, e^{\Delta L^n} H_i).
\]

Finally by using \( \Delta L^n = \tilde{L}^0_n - L^0_n \) we obtain

\[
H_i = Y_n + \sum_{j=1}^{s} \sum_{m=0}^{M} Z^{m,n}_{i,j} e^{-\Delta L^n} \tilde{g}_m(t_n + c^{n,j}_0, e^{\Delta L^n} H_j),
\]

\[
e^{L^0_n} V^0_n = e^{\Delta L^n} \left[ Y_n + \sum_{i=1}^{s} \sum_{m=0}^{M} c^{m,n}_i e^{-\Delta L^n} \tilde{g}_m(t_n + c^{n,i}_0, e^{\Delta L^n} H_i) \right],
\]

which is identical to Algorithm [1] given in (13), and consequently it holds that \( Y_{n+1} = e^{L^0_n} V^0_{n+1} \). \( \square \)

It follows that the (local) Lawson scheme converges:

**Theorem 1** (Strong convergence of Algorithm [1]). Let Assumption [2] hold, let \( X \) be the solution of SDE (1) and \( Y_n \) be the result of applying Algorithm [7] with an underlying SRK method of mean square order \( p \). Assume further that \( \Delta W^n_m = W_m(t_{n+1}) - W_m(t_n) \) in (12). Then there exists a \( \bar{c} \in \mathbb{R} \) such that for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \) it holds that

\[
E\|Y_N - X(t_N)\|_2 \leq \bar{c} h^p.
\]

**Proof.** This is proven by combining Lemmas [2] and [4] (choosing \( \tilde{L}^0_n = \sum_{i=0}^{n-1} \Delta L^i \)). \( \square \)

**Theorem 2** (Weak convergence of Algorithm [1]). Let Assumption [7] hold and assume that \( X \) is the solution of SDE (1). Moreover, assume that

(a) \( A_m \) for \( m > 0 \) are skew-symmetric,

(b) \( W^n \) and \( V^n_{n+1} \) are the numerical approximations of \( W(t_n) \) and \( \bar{X} \) obtained by the SRK method (12),

(c) this approximation is of weak order \( \bar{p} \), i.e. for all \( g \in C_\bar{p}(\mathbb{R}^{M+1} \times \mathbb{R}^d, \mathbb{R}) \) there exists a \( c \in \mathbb{R} \) such that for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \) it holds that

\[
|E(g(W^n, V^n_{n+1})) - g(W(t_n), V^0(t_n))| \leq c h^{\bar{p}},
\]

(d) \( W^0 = W_0(t_n) = t_n \).

Then for all \( f \in C_\bar{p}(\mathbb{R}^d, \mathbb{R}) \) there exists some constant \( c_f > 0 \) such that the numerical approximation \( Y_{n+1} = e^{\Delta L^n} V^n_{n+1} \) satisfies for all \( N \in \mathbb{N} \) and all \( n \in \{0, 1, \ldots, N\} \)

\[
|E(f(Y_n)) - E(f(X(t_n)))| \leq c_f h^{\bar{p}}.
\]

**Proof.** This is proven by combining Lemmas [3] and [4] (and choosing \( \tilde{L}^0_n = \sum_{i=0}^{n-1} \Delta L^i \)). \( \square \)

We want to emphasize that for Lipschitz continuous \( f \), the weak convergence order is at least as big as the strong order of convergence. So in the case where not all \( \{A_m\}_{m>0} \) are skew-symmetric, we still have weak convergence of at least the same order as the strong
order. However, this is not necessarily of the same order as the weak order of the underlying SRK scheme.

For the convergence of exponential schemes that do not originate from a Lawson type transformation, we refer to [11]. The convergence of SRK schemes is rather well studied and we refer to e.g. [8, 26, 12, 27] for the analysis of the mean square convergence order and [21, 9, 25, 12, 28] for the analysis of the weak convergence order.

In the following, we present some numerical examples showing the expected order of convergence.

3.2. Numerical example. In this section, we numerically show the strong ($p$) and weak ($\tilde{p}$) order of convergence of the following schemes:

- The Euler–Maruyama DSL scheme (“EM DSL”), [14] with $A_m = 0$ for $m > 0$, with expected strong and weak convergence orders $p = 0.5$ and $\tilde{p} = 1$.
- The Platen DSL scheme (“Platen DSL”), defined in (15) for $M = 1$, with $A_m = 0$ for $m > 0$, with expected strong and weak convergence orders $p = 1$ and $\tilde{p} = 1$.
- The Midpoint FSL scheme (“Midpoint FSL”), [16], with expected strong and weak convergence orders $p = 1$ and $\tilde{p} = 1$. In the numerical experiments, the solution of the implicit equation will be approximated with one single Newton step, evaluating the Jacobian once per time step, which has been proved sufficient to maintain the correct order, see [12]. For the below example, we also tested numerically that the results are not influenced significantly by doing more iterations.
- The Platen strong order 1.5 DSL scheme (“Platen 1.5 DSL”), defined in Section 7.1 for $M = 1$, with $A_m = 0$ for $m > 0$, with expected strong convergence order $p = 1.5$. For this scheme, we only demonstrate numerically the order of strong convergence.
- The Platen weak order 2.0 DSL scheme (“Platen 2.0 DSL”), defined in Section 7.2 for $M = 1$, with $A_m = 0$ for $m > 0$, with expected weak convergence order $p = 2.0$. For this scheme, we only demonstrate numerically the order of weak convergence.

For our numerical investigations we will use the mildly stiff Itô SDE

\[ dX = \begin{bmatrix} -\lambda & 0 \\ 0 & -\lambda \end{bmatrix} X + \begin{bmatrix} 0 & U_0(X) \\ -U_0(X) & 0 \end{bmatrix} X \, dt + \begin{bmatrix} 0 & 0.2 \\ -0.2 & 0 \end{bmatrix} X \, dW, \]

with $X(0) = (1, 0)^T$, $U_0(X) = \frac{1}{2}(X_1 + X_2)^2$, $t \in [0, 1]$ and $\lambda \in \{1, 5\}$. This corresponds to a non-linear oscillator [11], Eq. (18) perturbed by a linear attractor. The Midpoint FSL scheme will be applied to the corresponding Stratonovich version. By Itô’s formula it follows that $d||X(t)||^2 = (\lambda - 0.04)||X(t)||^2 \, dt$ and thus $||X(t)||_2 = e^{(\lambda - 0.04)t} \to 0$ for $t \to \infty$ if $\lambda > 0.2$. In all cases we choose $A_0 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ and run the numerical experiments in MATLAB version R2020a with 20 single-threaded workers on a 4x6-core Xeon 2.67 GHz Linux (Ubuntu 18.04) machine with 256 GB memory. The MATLAB source code for all numerical experiments in this section can be found at [14].

3.2.1. Strong convergence. We simulate 40 batches of each 25 paths for SDE (24) and compare the results to a numerical solution obtained by the Platen 1.5 DSL scheme with step size $h = 2^{-24}$.

For step sizes $h \in \{2^{-13}, 2^{-14}, 2^{-15}, 2^{-16}, 2^{-17}, 2^{-18}\}$ we report the average errors for $\lambda = 1$ and $\lambda = 5$ in Figs. [1a and 1b]. The 95% confidence intervals have been calculated and span in all cases less than $\pm 6.1\%$ of the corresponding error value.

In Fig. [1a] we see that for $\lambda = 1$ the Euler–Maruyama DSL scheme has an approximate strong order $p = 0.5$, the Platen DSL and Midpoint FSL schemes have an approximate strong order $p = 1.0$, and the Platen 1.5 DSL scheme has an approximate strong order of 1.5 as
Figure 1. Strong convergence results for SDE (24)
expected. Of the two first order schemes, the Midpoint FSL scheme has a smaller error than the Platen DSL, which is also expected as this scheme, in addition to the drift term, also incorporates the diffusion term into the exponential operator.

In Fig. 1b we observe that for $\lambda = 5$ the errors of the SL schemes are considerably smaller than the ones of their underlying methods. The order is as expected for all methods except the EM method. For the latter the order 1 error term dominates the order 0.5 term for the given step sizes, and we observe that the error more or less completely overlaps with the one of Platen’s method. We still notice that the error of EM DSL is smaller than the one of EM, although the measured order is smaller. For step sizes less than $2^{-18}$, we have observed that the error difference between EM and EM DSL is insignificant, and both methods exhibit order 0.5.

In Figs. 1c and 1d we depict the computational effort, measured as wall-clock time per batch of 25 paths, vs. the strong error averaged over all batches. Between the two Euler-Maruyama methods and the two Midpoint methods, respectively, a difference in efficiency can only be seen for $\lambda = 5$, while the DSL Platen methods are more efficient than their counterparts for both values of $\lambda$. The increased efficiency of the DSL-methods is more pronounced for $\lambda = 5$. For larger step sizes and especially $\lambda = 5$, similar to the above discussion it can be observed that the error of both the Euler-Maruyama and the Midpoint method is dominated by higher order terms.

3.2.2. Weak convergence. We simulate 2400 batches of each 15000 paths of SDE(24) and compare the results to a numerical solution obtained by the Platen 1.5 DSL scheme with step size $h = 2^{-14}$ as reference solution.

In Figs. 2a and 2b we report for step sizes $h \in \{2^{-9}, 2^{-8}, 2^{-7}, 2^{-6}, 2^{-5}\}$ the error when approximating $E(X_1^n)$ for $\lambda = 1$ and $\lambda = 5$. The 95%-confidence intervals have been calculated and span in all cases less than 6% of the corresponding error value, except for the Platen 2 DSL, where they span 10%, 14%, 33%, 68%, and 122% for $\lambda = 1$ and 2%, 6%, 25%, 98%, and 359% for $\lambda = 5$ for step-sizes $h = 2^{-5}, 2^{-6}, 2^{-7}, 2^{-8}$, and $2^{-9}$ respectively, as well as the EM DSL scheme in the case of $\lambda = 5$, where the confidence intervals span up to 27% of the corresponding error value.

We see that in this example, the EM DSL scheme and the Platen DSL scheme deliver very similar results. The same holds for the underlying schemes. For both the EM, Platen and Platen 2 schemes, the error of the DSL versions is much smaller than the one of the underlying scheme. All schemes show clearly the expected order of convergence, except that for $\lambda = 5$ and the considered step sizes, the Platen DSL scheme shows a somewhat smaller numerical order, and the Platen 2 DSL scheme exhibits one order higher than expected, caused by the order three dominating the order two error term.

In Figs. 2c and 2d we depict the wall-clock time per batch of 15000 paths vs. the weak error averaged over all batches. Also here, the results for the EM DSL scheme and the Platen DSL scheme are very similar, and the same holds for the underlying schemes. The DSL schemes are in all cases more efficient than their underlying methods.

For an application to highly oscillatory differential equations formulated as Stratonovich SDEs, we refer to [13]. We will in the following discuss linear stability for the Euler–Maruyama DSL and Platen DSL schemes.

4. Linear mean-square stability analysis

In this section, we calculate the mean-square stability regions of the Euler–Maruyama DSL and Platen DSL schemes based on the standard linear test equation [16]. Moreover we consider a higher-dimensional test-equation suggested by Buckwar and Kelly [4, 6, 5].
Weak convergence result for $\lambda = 1$: error of approximating $E[X_1^2]$ vs. step size

Wall-clock time $[s]$ vs. accuracy for $\lambda = 1$

Figure 2. Weak convergence results for SDE (24)
We assume that for all the methods considered, $\Delta W^n_m$ is chosen as exact Wiener increment, $\Delta W^n_m = W_m(t_{n+1}) - W_m(t_n)$.

4.1. **Linear stability for multiplicative noise.** To analyze linear stability in the case of multiplicative noise we consider the test equation

\begin{align}
X(t) &= X_0 + \int_0^t (\lambda + \sigma) X(s) \, ds + \int_0^t \mu X(s) \, dW(s),
\end{align}

with $X_0$ independent from $W, \lambda, \mu, \sigma \in \mathbb{C}$ and exact solution

\begin{align}
X(t) &= X_0 e^{(\lambda + \sigma - \frac{\mu^2}{2}) t + \mu W(t)},
\end{align}

which implies

$$\mathbb{E}(|X(t)|^2) = e^{(2\Re(\lambda + \sigma) + |\mu|^2)t} \mathbb{E}(|X_0|^2).$$

Thus, the solution of test equation (25) is mean square stable,

$$\lim_{t \to \infty} \mathbb{E}(|X(t)|^2) = 0,$$

if and only if

\begin{align}
2\Re(\lambda + \sigma) + |\mu|^2 < 0,
\end{align}

see, e.g., [29, 16].

4.1.1. **Analysis for Euler–Maruyama DSL method.** Application of the Euler–Maruyama DSL method with $A_0 = \lambda$ (and $A_1 = 0$, $g_0(t, x) = \sigma x$, $g_1(t, x) = \mu x$) to (25) yields

\begin{align}
Y_{n+1} &= e^{jh} (1 + h\sigma + \mu \Delta W^n) Y_n,
\end{align}

and therefore

\begin{align}
\mathbb{E}\left(|Y_{n+1}|^2\right) &= e^{2h\Re(\lambda)} \left(|1 + h\sigma|^2 + h|\mu|^2\right) \mathbb{E}\left(|Y_n|^2\right) \\
&= R(h\Re(\lambda), h\sigma, \sqrt{h}|\mu|) \mathbb{E}\left(|Y_n|^2\right)
\end{align}

with stability function

$$R : \mathbb{R} \times \mathbb{C} \times \mathbb{R}_+ \to \mathbb{R}_+ : R(z, u, v) = e^{2z}(|1 + u|^2 + v^2),$$

and a four-dimensional mean-square stability domain

\begin{align}
S = \{(z, u, v) \in \mathbb{R} \times \mathbb{C} \times \mathbb{R}_+ : R(z, u, v) \leq 1\},
\end{align}

which for an A-stable method would be a superset of the mean-square stability domain of the exact solution,

$$S^* = \{(z, u, v) \in \mathbb{R} \times \mathbb{C} \times \mathbb{R}_+ : 2z + 2\Re u + v^2 < 0\}.$$

For $(z, u, v) \in S^*$ and $|u| \leq -\sqrt{2} z$ (implying $z \leq 0$) it follows that

$$|1 + u|^2 + v^2 < 1 + |u|^2 - 2z \leq e^{-2z},$$

and thus $(z, u, v) \in S$. So, we can conclude that if the SDE has a mean-square stable solution (i.e. fulfills (27)) and fulfills

\begin{align}
|\sigma| \leq -\sqrt{2} \Re(\lambda),
\end{align}

then the Euler–Maruyama DSL scheme is mean-square stable independent of $h$. 
Note also that the domain of mean-square stability of the scheme is, as the one of the exact solution, not dependent on \( \mathcal{S} \), in contrast to the situation for the conventional Euler–Maruyama method, where the stability condition would read

\[ |1 + h(\sigma + \lambda)|^2 + h|\mu|^2 < 1. \]

4.1.2. Analysis for Platen DSL scheme. Application of the Platen DSL scheme to \((25)\) yields

\[
H_2 = (1 + h\sigma + \sqrt{h}\mu)Y_n,
\]

\[
Y_{n+1} = e^{4h} \left( Y_n + h\sigma Y_n + \mu\Delta W^n Y_n + \frac{\mu(H_2 - Y_n)}{\sqrt{h}} \left( \frac{(\Delta W^n)^2 - h}{2} \right) \right)
\]

and therefore

\[
E\left( |Y_{n+1}|^2 \right) = e^{2h\mathcal{R} \lambda} \left( |1 + h\sigma|^2 + h|\mu|^2 + \frac{h|\mu|^2}{2} |\sigma h + \mu \sqrt{h}|^2 \right) E\left( |Y_n|^2 \right)
\]

\[
= R(h \mathcal{R} \lambda, h\sigma, \sqrt{h}\mu) E\left( |Y_n|^2 \right),
\]

with stability function

\[
R : \mathbb{R} \times \mathbb{C} \times \mathbb{C} \rightarrow \mathbb{R}_+ : R(z, u, v) = e^{2z\left( |1 + u|^2 + |v|^2 \left( 1 + \frac{1}{2} |u + v|^2 \right) \right)}
\]

and now a five-dimensional mean-square stability domain

\[(30) \quad S = \{(z, u, v) \in \mathbb{R} \times \mathbb{C} \times \mathbb{C} : R(z, u, v) \leq 1 \}.
\]

So, in contrast to the Euler–Maruyama DSL, now the mean-square stability of the method depends also on the argument of \( \mu \), though still being independent of \( \mathcal{S} \) (in contrast to the conventional Platen scheme).

For \((z, u, v) \in \hat{S}^*\), where \( \hat{S}^* \) is the canonical embedding of \( S^* \) into \( \mathbb{R} \times \mathbb{C}^2 \), \(|u|^2 + |v|^2 \leq 2z^2\), \( \mathcal{R}(u\bar{v}) \leq 0 \) and \(|v|^2|u|^2 \leq -\frac{3}{2}z^3\) it follows that

\[
|1 + u|^2 + |v|^2(1 + \frac{1}{2}|u + v|^2) < 1 + |u|^2 + \frac{1}{2}|v|^2 \quad \frac{|u + v|^2}{(|u|^2 + |v|^2 + 2\mathcal{R}(u\bar{v}))} - 2z
\]

\[
= 1 - 2z + |u|^2 + \frac{1}{2}|v|^4 + \frac{1}{2} |v|^2|u|^2 + |v|^2 \mathcal{R}(u\bar{v}) \quad \leq 2z^2 \quad \leq -\frac{3}{2}z^3 \quad \leq 0
\]

\[
\leq e^{-2z},
\]

and thus \((z, u, v) \in S\). So, we can conclude that if the SDE has a mean-square stable solution (i. e. fulfills \((27)\) and fulfills

\[(31) \quad |\sigma|^2 + \frac{|\mu|^4}{2} \leq 2(\mathcal{R}(\lambda))^2, \quad \mathcal{R}(\sigma\bar{\mu}) \leq 0, \quad \text{and} \quad |\mu|^2|\sigma|^2 \leq -\frac{8}{3}(\mathcal{R}(\lambda))^3,
\]

then the Platen DSL scheme is mean-square stable independent of \( h \).

Overall we have however to note that the conditions \((29)\) and \((31)\) are by no means necessary but only sufficient.
4.2. Linear system stability for multiplicative noise. To study the linear system stability, we will follow the ideas outlined by Buckwar, Kelly and Sickenberger [7,5,4]. In summary: Given the linear SDE

\[ dX(t) = A_0 X(t) dt + \sum_{m=1}^{M} B_m X(t) dW_m \]

with \( A_0, B_1, \ldots, B_M \in \mathbb{R}^{d,d} \). Let \( P(t) = \mathbb{E} \text{vec}(X(t)X(t)^\top) \) be the expectation of the vectorization of the matrix process \( X(t)X(t)^\top \). Then \( P(t) \) is given by the solution of the \( d^2 \)-dimensional linear ODE

\[ dP(t) = SP(t) dt \]

where the mean-square stability matrix \( S \) for this system is given by [7]

\[ S = I_d \otimes A_0 + A_0 \otimes I_d + \sum_{m=1}^{M} B_m \otimes B_m, \]

where \( \otimes \) is the Kronecker product and \( I_d \) denotes the \( d \)-dimensional unit matrix. The zero solution \( X(t) = 0 \) is asymptotically mean-square stable if and only if all the eigenvalues of \( S \) have a negative real part, [7 Lemma 3.3]. Similarly, let \( Y_n \) be the numerical solution obtained by applying a one-step method to (32), such that

\[ Y_{n+1} = \mathfrak{B}_n Y_n, \]

and let \( Q_n = \mathbb{E} \text{vec}(Y_n Y_n^\top) \) be the expectation of the vectorization of the matrix process \( Y_n Y_n^\top \). Then

\[ Q_{n+1} = \mathfrak{S} Q_n, \quad \text{with} \quad \mathfrak{S} = \mathbb{E} \mathfrak{B}_n \otimes \mathfrak{B}_n, \]

and the method is asymptotically mean-square stable if and only if \( \rho(\mathfrak{S}) < 1 \) [7 Lemma 3.4].

Applying the Euler–Maruyama DSL scheme to (32), then a single step is given by

\[ Y_{n+1} = e^{A_0 h} \mathfrak{Y}_n, \quad \mathfrak{Y}_n = I_d + \sum_{m=1}^{M} B_m \Delta W_m. \]

Let \( \tilde{A} = A_0 h \) and \( \tilde{B}_m = B_m \sqrt{h} \), then it follows from the above that the stability matrix is given by

\[ \mathfrak{S} = e^{\tilde{A}} \otimes e^{\tilde{A}} \left( (I_d \otimes I_d) + \sum_{m=1}^{M} (\tilde{B}_m \otimes \tilde{B}_m) \right). \]

Similarly, applying the Platen DSL scheme to (32) with \( M = 1 \), we can write the one-step method as

\[ Y_{n+1} = e^{A_0 h} \mathfrak{Y}_n, \quad \text{with} \quad \mathfrak{Y}_n = \mathcal{I}_d + B_1 \Delta W^1_m + \frac{1}{2 \sqrt{h}} \left( e^{-A_0 h} B_1 e^{A_0 h} (\mathcal{I}_d + B_1 \sqrt{h}) - B_1 \right) \left( (\Delta W^1_m)^2 - h \right). \]

Using the same definitions for \( \tilde{A} \) and \( \tilde{B} \) as above the stability matrix of the scheme becomes

\[ \mathfrak{S} = e^{\tilde{A}} \otimes e^{\tilde{A}} \left( (I_d \otimes I_d) + (\tilde{B} \otimes \tilde{B}) + \left( e^{-\tilde{A}} \tilde{B} e^{\tilde{A}} (I_d + \tilde{B}) - \tilde{B} \right) \otimes \left( e^{-\tilde{A}} \tilde{B} e^{\tilde{A}} (I_d + \tilde{B}) - \tilde{B} \right) \right). \]

\[ ^{1}\text{The vectorization of a } d \times d \text{ matrix } A = \{a_{i,j}\}_{i,j=1}^{d} \text{ is the } d^2 \text{-dimensional vector given by vec}(A) = (a_{1,1}, a_{2,1}, \ldots, a_{d,d}, a_{d,d})^\top. \]
or with $\mathcal{C} := e^{-\hat{A}B}e^{\hat{A}(I_d + B)} - B$

$$\mathcal{S} = e^{\hat{A}} \otimes e^{\hat{A}} \left( (I_d \otimes I_d) + (\hat{B} \otimes \hat{B}) + (\hat{C} \otimes \hat{C}) \right).$$

The methods are asymptotically stable if $\rho(\mathcal{S}) < 1$. Note that if $\hat{A}$ and $\hat{B}$ commute, then $\mathcal{C}$ reduces to $\mathcal{C} = \hat{B}^2$, which corresponds to the term expected for the classical Platen scheme, whose stability matrix is

$$\mathcal{S} = (I_d + \hat{A}) \otimes (I_d + \hat{A}) + (\hat{B} \otimes \hat{B}) + (\hat{B}^2 \otimes \hat{B}^2).$$

So in contrast to the relation between the stability regions of the standard Euler–Maruyama and the Euler–Maruyama DSL schemes, the commutator $[\hat{A}, \hat{B}]$ influences the relation between the standard Platen and the Platen DSL schemes.

We will now calculate and plot the stability regions for some examples and also show some corresponding SDE simulations. We will compare three schemes:

- the Euler–Maruyama DSL scheme (“EM DSL”) [14],
- the Platen DSL scheme (“Platen DSL”) [15],
- the implicit Platen strong order 1.0 scheme (“Impl. Platen”) [18] Chapter 12.3.1.

4.2.1. Linear test equation with orthogonal noise. We consider the linear test equation with noise orthogonal to the flow $(X_1(t), X_2(t))$ [5] Equ. 9]

$$d\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} = \begin{pmatrix} \lambda & b \\ 0 & \lambda \end{pmatrix} \begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} dt + \begin{pmatrix} 0 & \sigma \\ -\sigma & 0 \end{pmatrix} \begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} dW(t)$$

where $\lambda, b, \sigma \in \mathbb{R}$. With $\hat{A} = hA_0$, $\hat{B} = \sqrt{h}B_1$ and $\hat{C}$ as defined above we can now calculate the stability matrices $\mathcal{S}(\lambda h, bh, \sigma \sqrt{h})$ according to [14] and [35] for the two Lawson schemes and according to (36) for the implicit Platen scheme. The mean square asymptotic stability region of the schemes is in each case given by $\{(\lambda h, bh, \sigma \sqrt{h}) \in \mathbb{R}^3 : \rho(\mathcal{S}(\lambda h, bh, \sigma \sqrt{h})) < 1\}$. In Fig. 3 we depict for various choices of $bh$ four slices of these stability regions and the corresponding stability region of the exact solution.

In Fig. 3 we see that the Euler–Maruyama DSL scheme has the best stability properties for the presented choices of $bh$, $\lambda h$ and $\sigma^2 h$, but in some cases might over-stabilise, in particular for small values of $bh$. Among the three schemes, the Platen DSL scheme reproduces the stability region of the exact solution the best, especially for smaller values of $bh$. The implicit Platen scheme shows in all cases a lacking satisfactory stabilising effect in $\sigma^2 h$, which is expected as this scheme is only implicit in the drift.

To demonstrate this effect, we now consider the situation that $b = 10$, $\lambda \in \{-20, -10\}$, $\sigma^2 = 25$ and $h = 0.1$ corresponding to the two points in the top right figure of Fig. 3. From this stability plot, we expect that for $\lambda h = -2.0$ and $\sigma^2 h = 2.5$ the two Lawson schemes perform better than the implicit Platen scheme, while for $\lambda h = -1.0$ we expect that the Euler–Maruyama DSL scheme over-stabilises and thus converges to 0, whereas the two Platen schemes diverge. We use the initial value $X_0 = (1, 1)^T$ and simulate $10^6$ paths.

In Fig. 4 we plot the evolution of the second moment of $X_1$ and $X_2$. To calculate the exact moments, we derived the ODE system for $E(X_1^2)$, $E(X_2^2)$ and $E(X_1X_2)$ using Itô’s formula and solved it in Matlab using ”ode15s” [30] with an absolute tolerance of $10^{-14}$. We see that both the Platen DSL and Euler–Maruyama DSL schemes are indeed stable just inside their stability region, $\lambda h = -2.0$ and $\sigma^2 h = 2.5$, whereas the implicit Platen scheme fails to be stable as predicted by its stability region. Conversely, we see that when $\lambda h = -1.0$ and $\sigma^2 h = 2.5$, then the implicit Platen scheme diverges due to the parameters lying outside of its stability region, while the Platen DSL scheme remains close to 1, indicating that
\[ b = 0 \]
\[ bh = 1 \]
\[ bh = 2 \]
\[ bh = 5 \]

**Figure 3.** Stability regions of the Euler–Maruyama DSL, Platen DSL and implicit Platen scheme applied to the linear test equation. The coloured areas show the direction where \( \rho(\mathbb{S}) < 1 \). The two marks mark the values for which we do the numerical simulation.

**Figure 4.** Numerical results for (37) on the boundary of the stability region of the exact solution. \((\lambda h, \sigma^2 h)\) is near the boundary of the scheme's stability region with \( \rho(\mathbb{S}) \) close to 1. The Euler–Maruyama DSL scheme converges to 0, as the chosen parameters are still inside of its stability region.
4.2.2. Damped and driven oscillators. In the above experiments, we saw that the implicit Platen scheme, in contrast to the Lawson schemes, might fail to stabilise the SDE when the stiffness comes from the diffusion. However, for the damped/driven oscillator, we will see that eigenvalues with large complex parts might make the implicit Platen over-stabilise.

We consider the oscillator

$$\begin{align*}
\frac{d}{dt} \begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} &= \begin{pmatrix} \lambda & \omega^2 \\ -\omega^2 & \lambda \end{pmatrix} \begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} dt + \begin{pmatrix} 0 & \sigma \\ -\sigma & 0 \end{pmatrix} \begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} dW(t) \\
&= A_0 = \begin{pmatrix} hA_0 \\ B_1 \end{pmatrix}
\end{align*}$$

with $\lambda, \omega, \sigma \in \mathbb{R}$. For this system, the matrices needed to compute the stability domains (34) to (36) are $A = hA_0$, $B = \sqrt{h}B_1$, and $C = -h\sigma^2 I_d$. We note that with $\lambda$ and $\omega$ we can control the size of the real and imaginary part of the eigenvalues respectively of the matrix $A_0$. Now plotting the stability regions, we obtain Fig. 5.

In Fig. 5 we see that, as for the scalar test equation (25), the stability domains of both Lawson schemes and the exact solution are independent of the imaginary part of the eigenvalues of $A_0$, i.e. $\omega$, whereas the one of the implicit Platen scheme depends on it.

To show the consequence, consider the oscillator with $X_0 = (1, 1)^T$, $\omega^2 = 10\pi$, $\sigma^2 = 1$, $\lambda \in \{-1, 0, 1\}$ and $h = 0.1$ corresponding to the two marks in the left part of Fig. 5. For $\lambda h = 0.1$ it is a driven oscillator, whereas for $\lambda h = 0.3$ it is a damped one.

We simulate $E(X(t))$ using the Euler–Maruyama DSL, Platen DSL and implicit Platen schemes with $h = 0.1$ and 10000 paths. We also include the exact moment which, again, is obtained by solving the ODE system for $E(X_1^2)$, $E(X_2^2)$ and $E(X_1X_2)$ in Matlab using "ode15s" with an absolute tolerance of $10^{-14}$.

In Fig. 6 we see that both for the damped oscillator and the driven oscillator, the implicit Platen scheme stabilises the numerical flow significantly more than the Platen DSL scheme. For the damped oscillator, the behaviour of all schemes is correct, and they all converge to 0. The implicit Platen scheme does, however, converge way too fast compared to the exact solution. The Euler–Maruyama DSL scheme is also converging slightly too fast, but is closer to the exact solution. The Platen DSL scheme follows almost the exact solution.
Damped oscillator: $\lambda h = -0.3$ and $\sigma^2 h = 0.4$.

Driven oscillator: $\lambda h = -0.1$ and $\sigma^2 h = 0.4$.

**Figure 6.** Numerical results for the driven and damped oscillators using the Euler–Maruyama DSL, the Platen DSL, and the implicit Platen schemes.

For the driven oscillator, we see that the implicit Platen scheme shows the wrong behaviour, as if the oscillator would still be damped. The two Lawson schemes behave correctly, with the results of the Platen DSL scheme visually matching the exact solution and the approximations by the Euler–Maruyama DSL scheme being slightly off.

5. Conclusion

In this paper, we derived the general class of SRK Lawson schemes. We proved that, if the underlying SRK scheme is of mean-square order $p$, then the SRK Lawson scheme is of strong order $p$. Similarly, under the assumption that the linear diffusion terms included in the exponential operator are skew-symmetric, we proved that the SRK Lawson schemes also inherit the weak order of convergence from the underlying SRK scheme.

We performed a linear stability analysis for the Euler–Maruyama and Platen DSL schemes. In particular, we numerically demonstrated that the implicit Platen scheme might provide insufficient stabilisation when the destabilisation comes from the diffusion. However, the Euler–Maruyama and Platen DSL schemes provide adequate stabilisation, even though the exponential only includes the drift term. Conversely, we demonstrate that for driven oscillators with small diffusion terms, the implicit Platen scheme might over-stabilise and make it a damped oscillator, whereas the two SL schemes more accurately match the behaviour of the exact solution.

6. Acknowledgement

Nicky Cordua Mattsson would like to thank the SDU e-Science centre for partially funding his PhD and the Department of Mathematics at the Norwegian University of Science and Technology for kindly hosting him during his visit. The authors would like to thank two anonymous reviewers for very detailed reading and the resulting many helpful comments.
7. Appendix

7.1. Platen strong order 1.5 SL scheme. Writing the explicit order 1.5 strong scheme by Platen for $M = 1$ [18, Eq. 11.2.1] in the form (10), we see that the coefficients $z_{i}^{m,n}$ of the scheme are given by

| $i$  | $m = 0$       | $m = 1$       |
|------|---------------|---------------|
| 1    | $z_{1}^{0,n} = \frac{1}{2} h$ | $z_{1}^{1,n} = \Delta W^n - \frac{1}{3} [\Delta W^n h - \Delta Z^n]$ |
| 2    | $z_{2}^{0,n} = \frac{1}{2 h} \Delta Z^n + \frac{1}{4} h$ | $z_{2}^{1,n} = \frac{1}{h} \left[ \frac{1}{2} \Delta W^n h - \Delta Z^n - \frac{1}{4} \left( \frac{1}{3} \Delta W^n - h \right) \Delta W^n \right]$ |
| 3    | $z_{3}^{0,n} = -\frac{1}{2 h} \Delta Z^n + \frac{1}{4} h$ | $z_{3}^{1,n} = \frac{1}{h} \left[ \frac{1}{2} \Delta W^n h - \Delta Z^n + \frac{1}{4} \left( \frac{1}{3} \Delta W^n + h \right) \Delta W^n \right]$ |
| 4    | $z_{4}^{0,n} = 0$ | $z_{4}^{1,n} = \frac{1}{4 h} \left[ \frac{1}{2} \left( \Delta W^n \right)^2 - h \right] \Delta W^n$ |
| 5    | $z_{5}^{0,n} = 0$ | $z_{5}^{1,n} = -\frac{1}{4 h} \left[ \frac{1}{3} \left( \Delta W^n \right)^2 - h \right] \Delta W^n$ |

where

$$\Delta Z^n = \frac{1}{2} h \Delta W^n + \frac{\sqrt{3}}{6} h^{3/2} U^n$$

with $U^n \sim N(0, 1)$. Similarly, the coefficients of $Z_{i,j}^{m,n}$ are given by
Finally, using the definitions of the coefficients $c_{m,i}^n$ and $\Delta L_i^n$ we calculate

| $m=0$ | $m=1$ | $\Delta L_i^n$ |
|-------|-------|----------------|
| $i=1$ | $c_{0,1}^{n,1} = 0$ | $c_{1,1}^{n,1} = 0$ | $\Delta L_1^n = 0$ |
| $i=2$ | $c_{0,2}^{n,2} = h$ | $c_{1,2}^{n,2} = \sqrt{h}$ | $\Delta L_2^n = (A_0 - \gamma A_1^2)h + A_1\sqrt{h}$ |
| $i=3$ | $c_{0,3}^{n,3} = h$ | $c_{1,3}^{n,3} = -\sqrt{h}$ | $\Delta L_3^n = (A_0 - \gamma A_1^2)h - A_1\sqrt{h}$ |
| $i=4$ | $c_{0,4}^{n,4} = 2\sqrt{h}$ | $c_{1,4}^{n,4} = 0$ | $\Delta L_4^n = (A_0 - \gamma A_1^2)h + 2A_1\sqrt{h}$ |
| $i=5$ | $c_{0,5}^{n,5} = h$ | $c_{1,5}^{n,5} = 0$ | $\Delta L_5^n = (A_0 - \gamma A_1^2)h$ |

Finally, using the definitions of $c_m^n$ and $\Delta L_i^n$ we calculate $c_0^n = h$, $c_1^n = \Delta W^n$ and thus $\Delta L_i^n = (A_0 - \gamma A_1^2)h + A_1\Delta W^n$.

With all the parameters of the scheme \{13\} in place, the resulting Platen strong order 1.5 SL scheme is given by

\begin{align*}
H_1 &= \bar{V}_n^n, \\
H_2 &= \bar{V}_n^n + \bar{g}_0^n(t_n, H_1)h + \bar{g}_1^n(t_n, H_1)\sqrt{h}, \\
H_3 &= \bar{V}_n^n + \bar{g}_0^n(t_n, H_1)h - \bar{g}_1^n(t_n, H_1)\sqrt{h}, \\
H_4 &= \bar{V}_n^n + \bar{g}_0^n(t_n, H_1)h + \bar{g}_1^n(t_n, H_1)\sqrt{h} + e^{-\Delta L_2^n/2}\bar{g}_1^n(t_n + \Delta W^n/H_2)\sqrt{h}, \\
H_5 &= \bar{V}_n^n + \bar{g}_0^n(t_n, H_1)h + \bar{g}_1^n(t_n, H_1)\sqrt{h} - e^{-\Delta L_2^n/2}\bar{g}_1^n(t_n + \Delta W^n/H_2)\sqrt{h}, \\
\end{align*}

\[
\bar{V}_n^{n+1} = \bar{V}_n^n + \bar{g}_1^n(t_n, H_1)\Delta W^n + \left[ e^{-\Delta L_2^n/2} \bar{g}_0^n(t_n + \Delta W^n/H_2) - e^{-\Delta L_2^n/2} \bar{g}_0^n(t_n + \Delta W^n/H_3) \right] \frac{\Delta Z^n}{2\sqrt{h}}
\]
Following the same steps as above, for the explicit weak scheme by Platen for $t = 1$ [13] as underlying scheme we obtain

\begin{align*}
\Delta L^n_1 &= 0 \\
\Delta L^n_2 &= (A_0 - \frac{1}{2} A^2_1) h + A_1 \Delta W^n \\
\Delta L^n_3 &= (A_0 - \frac{1}{2} A^2_1) h + A_1 \sqrt{h} \\
\Delta L^n_4 &= (A_0 - \frac{1}{2} A^2_1) h - A_1 \sqrt{h}
\end{align*}

and thus the corresponding Platen weak order 2.0 SL scheme is given by

\begin{align*}
Y_{n+1} &= e^{\Delta L^n_1} \overline{V}^n_{n+1} \\
H_1 &= \overline{V}^n_n \\
H_2 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h + \overline{g}^n_1 (t_n, H_1) \Delta W^n \\
H_3 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h + \overline{g}^n_1 (t_n, H_1) \sqrt{h} \\
H_4 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h - \overline{g}^n_1 (t_n, H_1) \sqrt{h}
\end{align*}

\begin{align*}
\overline{V}^n_{n+1} &= \overline{V}^n_n + \left[ e^{-\Delta L^n_2} \overline{g}^n_0 (t_n + h, e^{\Delta L^n_2} H_2) + \overline{g}^n_0 (t_n, H_1) \right] \frac{h}{2} \\
&\quad + e^{-\Delta L^n_2} \overline{g}^n_1 (t_n + h, e^{\Delta L^n_2} H_3) + 2 \overline{g}^n_1 (t_n, H_1) + e^{-\Delta L^n_2} \overline{g}^n_1 (t_n + h, e^{\Delta L^n_2} H_4) \frac{\Delta W^n}{4} \\
&\quad + e^{-\Delta L^n_2} \overline{g}^n_1 (t_n + h, e^{\Delta L^n_2} H_5) - e^{-\Delta L^n_2} \overline{g}^n_1 (t_n + h, e^{\Delta L^n_2} H_6) \frac{(\Delta W^n)^2 - h}{4 \sqrt{h}}
\end{align*}

7.2. Platen weak order 2.0 SL scheme. Following the same steps as above, for the explicit order 2 weak scheme by Platen for $M = 1$ [13] Eq. 15.1.1] as underlying scheme we obtain

\begin{align*}
\Delta L^n_1 &= 0 \\
\Delta L^n_2 &= (A_0 - \frac{1}{2} A^2_1) h + A_1 \Delta W^n \\
\Delta L^n_3 &= (A_0 - \frac{1}{2} A^2_1) h + A_1 \sqrt{h} \\
\Delta L^n_4 &= (A_0 - \frac{1}{2} A^2_1) h - A_1 \sqrt{h}
\end{align*}

and the corresponding Platen weak order 2.0 SL scheme is given by

\begin{align*}
Y_{n+1} &= e^{\Delta L^n_1} \overline{V}^n_{n+1} \\
H_1 &= \overline{V}^n_n \\
H_2 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h + \overline{g}^n_1 (t_n, H_1) \sqrt{h} \\
H_3 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h + \overline{g}^n_1 (t_n, H_1) \sqrt{h} \\
H_4 &= \overline{V}^n_n + \overline{g}^n_0 (t_n, H_1) h - \overline{g}^n_1 (t_n, H_1) \sqrt{h}
\end{align*}