Attempts at a numerical realisation of stochastic
differential equations containing Preisach operator

S McCarthy\textsuperscript{1} and D Rachinskii\textsuperscript{1}
\textsuperscript{1}Department of Applied Mathematics, University College Cork, Cork, Ireland
E-mail: stephen.p.mccarthy@umail.ucc.ie

Abstract. We describe two Euler type numerical schemes obtained by discretisation of a
stochastic differential equation which contains the Preisach memory operator. Equations of
this type are of interest in areas such as macroeconomics and terrestrial hydrology where
deterministic models containing the Preisach operator have been developed but do not fully
encapsulate stochastic aspects of the area. A simple price dynamics model is presented as one
motivating example for our studies. Some numerical evidence is given that the two numerical
schemes converge to the same limit as the time step decreases. We show that the Preisach
term introduces a damping effect which increases on the parts of the trajectory demonstrating
a stronger upwards or downwards trend. The results are preliminary to a broader programme
of research of stochastic differential equations with the Preisach hysteresis operator.

1. Introduction
A good body of work has been done of developing, studying and solving deterministic
differential models which contain the Preisach operator. The simplest models consist of one
scalar differential equation coupled with the Preisach operator input-output relationship. In
particular, these types of models have been used in the context of terrestrial hydrology of
homogeneous soil water systems with the Preisach operator introducing a hysteretic constitutive
relationship between the moisture content and the pressure (matric potential) to the ordinary
differential equation obtained by averaging out the spatial variation from the Philip-Richards
balance equation [1–3]. Another example is provided by models of macroeconomic systems and
multiagent market models where the Preisach operator is used to incorporate shock type memory
into the system [4,5]. Both of these areas contain a stochastic nature which is not fully captured
by deterministic models.

Stochastic aspects of the input-output relationship defined by the Preisach operator have been
fairly well understood in the case of open loop systems where the properties of the stochastic
input are known or measurable \textit{a priori} [6]. The formalism of the phenomenological Preisach
model presents a system as a weighted superposition of many two-state non-ideal relays that
are individually and independently driven by the same input [7,8]. As this formalism is rather
general in modelling rate-independent systems exhibiting hysteresis and shock type memory,
the theory of a stochastically driven Preisach operator has been motivated by, and successfully
applied to, a variety of problems such as modelling thermal relaxation and viscosity (after-
effect) in ferromagnetic materials [9,10], creep in superconductors [11], signal processing and
passage of noise through hysteretic systems [12], effect of noise on data storage technologies and
data collapse [13] and others. Differential models of closed loop systems involving the Preisach operator have received less attention. The existing body of results refers mainly to systems with piecewise smooth trajectories such as models of stochastically driven mechanical systems, shape memory alloys and hysteretic oscillators [14]; estimation of damage and fatigue [15]; and, models of terrestrial hydrology [16].

As a natural stochastic extension of the deterministic equations with the Preisach operator considered in [3–5] in the context of modelling hydrological systems and economics systems, we propose the following formal stochastic differential equation

\[ \xi dx_t + d(Px)_t = a(x_t, t)dt + b(x_t, t)dW_t \]  

where \( P \) denotes the Preisach operator, i.e., \( (Px)_t = (Px)(t) \) denotes output of the Preisach model with continuous input \( x_t = x(t) \); \( \xi \) is a positive parameter; \( W_t \) is the Wiener process; and, \( a(\cdot, \cdot), b(\cdot, \cdot) \) are continuous functions. This extension may be of interest in the areas of hydrology; economics and finance where stochastic effects are prevalent. As a prototype example, we will consider below a simple price dynamics model leading to a discrete time counterpart of equation (1) with constant \( a \) and \( b \). This price dynamics model adopts, yet in an essentially simplified form, the philosophy of the modelling approach proposed in [17–21] which leads to a hierarchy of powerful multiagent models of economics systems.

Equation (1) can also be used for examining the effect of feedback on various open loop systems modelled by the Preisach operator and by the inverse Preisach operator with a stochastic input, as well as for testing the effect of noise on deterministic differential models involving the shock type memory.

We note that the simplest example of the shock type memory is the running maximum (running minimum) of the input which is one of the standard market indicators in financial setting. Stochastic differential equations involving running extrema have been studied, for instance, in [22]. From this perspective, equation (1) can be viewed as a similar model incorporating a more complex memory of the past shock type events (called the main extrema of the input in [23]). The role of such shocks in economics systems is discussed in [4]. Equation (1) is thus the simplest model incorporating both the Preisach memory operator and the stochastic component.

We stress that the mathematical formalism of the continuous time system (1) is not the subject of this paper. We will consider and compare two natural Euler type formal discretisation of equation (1). Some numerical evidence will be given that the two discrete time schemes converge as the time step becomes smaller. We will then discuss a damping effect introduced by the Preisach memory term to the system. A characteristic feature of this effect is that damping is stronger on the parts of a trajectory that demonstrate a more pronounced upward or downward trend.

The paper is organized as follows. In the next section, we briefly remind the construction of the Preisach operator introducing the necessary notation. In Section 3, we discuss a price dynamics model leading to a discrete counterpart of equation (1). In Section 4, two formal discretisations of equation (1) are presented. Section 5 contains results of implementation of the two proposed numerical schemes and discussion. The last section presents conclusions.

The results presented here are preliminary to a programme of future work which will aim at (a) developing mathematical formalism of the continuous time equation (1); (b) developing numerical methods of solving equation (1) and analysis of convergence of the discrete time schemes to the continuous limit; (c) improving the efficiency and accuracy of the numerical schemes (using, in particular, the Brownian bridge techniques); (d) adapting analytical tools of the theory of stochastic differential equations, such as the Fokker-Planck equation etc., to equation (1) wherever possible; and, (e) examining the long term effect of stochastic inputs on the Preisach memory configuration in closed loop systems. This work will be done elsewhere.
2. Preisach model
Here we detail some of the properties of the Preisach operator as they pertain to the work examined in this paper.

2.1. Non-ideal relay
The non-ideal relay is a basic hysteresis element. The dynamics of the relay are shown in figure 1. The state of the relay can be either 0 (the relay is switched off) or 1 (the relay is switched on) at any instant (sometimes, the states ±1 are used instead with a straightforward change in all the definitions). The dynamics of the state is defined by the equation

\[
y(t) = R_{\alpha,\beta}[\eta_0]x(t) = \begin{cases} 
1 & \text{if } \exists t_1 \in [t_0, t] \text{ such that } x(t_1) \geq \beta, x(\tau) > \alpha \forall \tau \in [t_1, t] \\
0 & \text{if } \exists t_1 \in [t_0, t] \text{ such that } x(t_1) \leq \alpha, x(\tau) < \beta \forall \tau \in [t_1, t] \\
\eta_0 & \text{if } \alpha < x(\tau) < \beta \forall \tau \in [t_0, t]
\end{cases}
\]

for \( t \geq t_0 \), where \( \eta_0 = y(t_0) \) is the initial state of the relay at time \( t_0 \). Hence, the state of the relay is uniquely determined after the moment \( t_0 \) for a known continuous input \( x(t) \), \( t \geq t_0 \), if the initial state \( \eta_0 \) is given.

2.2. The Preisach Operator
The Preisach operator can be thought of as a weighted sum of independent non-ideal relays with a common input. The threshold values of the relays are represented by points in the Preisach half-plane, \( \Pi = \{(\alpha, \beta) : \beta > \alpha \} \). The operator maps a continuous input \( x(t) \), \( t \geq t_0 \), to the continuous output

\[
y(t) = P[\eta_0(\alpha, \beta)]x(t) = \int_{\alpha<\beta} \mu(\alpha, \beta)R_{\alpha,\beta}[\eta_0(\alpha, \beta)]x(t)d\alpha d\beta \quad t \geq t_0
\]

of the Preisach model. Here the integrable nonnegative function \( \mu(\alpha, \beta) \) called the Preisach weight function describes the weighting of the non-ideal relays; the binary function \( \eta_0 = \eta_0(\cdot, \cdot) \) called the initial state of the Preisach operator is an infinite-dimensional parameter describing the states of all the relays at the initial moment \( t_0 \). We will also adopt a shorter notation \( y(t) = (Px)(t) \) for the Preisach operator omitting the reference to the initial state parameter \( \eta_0 = \eta_0(\alpha, \beta) \) wherever it does not lead to a confusion.

Similarly, a discrete version of the Preisach operator can be defined, where the number of relays is finite, the integral in (3) is replaced with summation, and the Preisach weight function is replaced with a finite set of nonnegative weights \( \mu_i \).

2.3. Geometrical interpretation
According to equation (3) the output of the Preisach operator is given by an integral over the Preisach half-plane. An efficient method of calculating the output is based on the following geometrical interpretation of the states of the Preisach operator, see, for example, [23] for details. Let us consider the area on the Preisach half-plane for which the relays are switched on at a given moment \( t \). In figure 2, this area lies to the left of the staircase line \( S(t) \), while the relays to the right of this line are switched off. Hence, the output value \( y(t) \) equals the integral of the Preisach weight function \( \mu(\alpha, \beta) \) over the area to the left of the line \( S(t) \). The evolution of the input results in the evolution of the line \( S(t) \) and thus in the evolution of the output \( y(t) \). The evolution of \( S(t) \) is defined by a few simple rules. We note that all the segments of the staircase line \( S(t) \) are parallel to the coordinate axes at every time instant; for systems with piecewise monotone inputs or discrete time inputs, which can be interpolated to piecewise
Figure 1. Non-ideal relay. Representation of the dynamics of the non-ideal relay with thresholds $\alpha$ and $\beta$, with $\beta > \alpha$.

Figure 2. The Preisach half-plane. The boundary line $S(t)$ separates the domain where the points $(\alpha, \beta)$ represent the relays which are switched on (below the line $S(t)$) from the domain where the relays are switched off (above the line $S(t)$) at a given moment $t$. The coordinates of the point $A$ equal $x(t)$, the input value.

2.4. Preisach memory state

The boundary line $S(t)$ is known as the state of the Preisach operator. Its position on the Preisach half-plane is defined by the location of its corners described by the coordinate pairs $(m_k, M_k)$. Figure 6 shows the projection of these corners onto the bisector $\beta = \alpha$. The dynamics of the input results in the deletion of certain corners and the creation of new ones and is thereby translated to the dynamics of the state and the output of the Preisach operator as described in the previous subsection. At a given moment $t$, the abscissas $m_0, m_1, \ldots, m_k$ of the corners of $S(t)$ form a subset of the set of all the local minima of the input $x(t)$ which have been achieved prior to the moment $t$. They are called the running main minima of the input, see [23] for the
definition and discussion. Likewise the ordinates $M_0, M_1, \ldots, M_k$ of the corners of $S(t)$ are a specific selection of the local maxima of the input which $x(t)$ has achieved prior to the moment $t$; they are called the running main maxima. The running main extrema (also known as past shock values of the input) at every instant are ordered as

$$
\cdots < m_k < \cdots < m_1 < m_0 < x(t) < M_0 < M_1 < \cdots < M_k < \cdots
$$

Figure 3. An example of creating a new corner in the line $S(t)$ at an extremum (turning) point of the input $x(t)$.

Figure 4. An example of erasing (wiping out) corners from the line $S(t)$.

Figure 5. Changes in the output of the Preisach operator. (a) The input increases from the value $x_1$ to the value $x_2$. As a result the relays in the shaded area switch on; other relays do not change their state. The output (3) increases by the integral of the Preisach weight function $\mu = \mu(\alpha, \beta)$ over the shaded area. (b) The input decreases from the value $x_1$ to $x_2$. The relays in the shaded area switch off. The output decreases by the integral of $\mu$ over the shaded area.

Figure 6 shows that given the same sequences of the main maxima and minima the state $S(t)$ can be arranged in two possible ways, one where the last link of $S(t)$ connecting the corner point $(m_0, M_0)$ with the point $(x(t), x(t))$ on the bisector $\alpha = \beta$ is horizontal and the other where the last link is vertical. In order to uniquely describe the state, we use an additional variable $\zeta$ which can have either the value 1 or $-1$. The value of 1 corresponds to the arrangement of the
state for which the last link is horizontal and the −1 value means that the last link is vertical. Thus, given the input value \( x(t) \) at a moment \( t \), the value of the variable \( \zeta \) and the two arrays \( \omega_m = \{m_0, m_1, \ldots\} \), \( \omega_M = \{M_0, M_1, \ldots\} \) containing the values of the running main extrema at this moment completely define the state \( S(t) \) (and thus the memory of the system at the moment \( t \)). This notation is used in Section 4 below where updating the state and output of the Preisach operator in response to changes of the input is implemented as part of the numerical schemes for solving equation (1).

![Diagrams](https://example.com/diagram.png)

**Figure 6.** The Preisach operator state with the corners \((m_k, M_k) = (m^n_k, M^n_k)\) at a moment \( t = n \) where the last link is (a) vertical, \( \zeta = -1 \); (b) horizontal, \( \zeta = 1 \). The arrays of the main extrema are \( \omega_m = \{m^n_0, m^n_1, \ldots\} \), \( \omega_M = \{M^n_0, M^n_1, \ldots\} \).

3. **A motivating example: A Price dynamics model**

Let us consider a model of a market of many traders. We model each single trader by a non-ideal relay with thresholds \( \alpha, \beta \) where \( \alpha < \beta \); the thresholds are different for different traders. Hence, a single trader has two possible states \( s = \pm 1 \). Let \( s = -1 \) be the ‘long’ state, i.e., the trader has purchased an asset at a price lower than the value of his \( \beta \) threshold and waits until the price goes up to the value of \( \beta \) before selling it with profit and switches to the state \( s = 1 \). The state \( s = 1 \) is ‘short’, the trader has not purchased an asset and waits for the price to fall to the level of his \( \alpha \) threshold at which he wants to buy the asset at this point switching to the state \( s = -1 \).

Now we consider a pool of \( M \) traders, with the \( i \)-th trader having thresholds \( \alpha_i, \beta_i \), and construct a discrete version of the Preisach model of them. If we take the input to the system to be the price \( p \) of the asset, then, following [20], the output

\[
\sigma = \frac{1}{M} \sum_{i=1}^{M} \mu_i s_i
\]

measures the sentiment of the market. Here \( s_i \) is the state of the \( i \)-th trader; \( \mu_i > 0 \) is his weight in forming public opinion about the value of the asset (possibly related to the amount of money or asset he has, etc.).

We assume the dynamics of the price of the asset to follow the equation from [20]. That is, the price evolves in discrete time as

\[
p_{t+h} = p_t \exp \left[ \eta \sqrt{h} + \left( a - \frac{1}{2} \right) h - \kappa \Delta \sigma_t \right]
\]
where $h$ is the time step; the sequence of independent Gaussian random quantities $\eta_t \sim N(0, 1)$ models the external random information stream; $a$ is the trend term; $-1/2$ correction term is the same as in the classical setting where it comes from Ito’s formula; $\kappa > 0$ is a parameter measuring the effect of the sentiment; and, $\Delta \sigma_t = \sigma_{t+h} - \sigma_t$ is the change in sentiment which creates a pressure on price. The minus sign in front of the hysteresis term $\kappa \Delta \sigma_t$ accounts for the fact that when more traders switch to the 'long' state this creates a positive pressure pushing the price up. Similarly, switching of more traders to the 'short' state creates a pressure on the price to move down. That is, there is a positive feedback loop. Note that setting $\kappa = 0$ transforms (5) to the discrete form of the standard geometric Brownian motion.

If we perform a logarithmic scaling of the price by introducing the variable $r = \ln \frac{p}{p_0}$ instead of $p$, then equation (5) can be rewritten as

$$\Delta r_t = \eta_t \sqrt{h} + \left( a - \frac{1}{2} \right) h - \kappa \Delta \sigma_t$$

(6)

with $\Delta r_t = r_{t+h} - r_t$. It should be noted that according to (4), $\sigma_t$ is the output of the discrete Preisach model with the input $p_t$. However, we can use $r_t$ as the input instead of $p_t$ if we also rescale the thresholds of the relays (or, equivalently, the Preisach measure density function). This is achieved by scaling the threshold values $\alpha_i, \beta_i$ of the $i$-th relay to instead be $\ln(\alpha_i/p_0), \ln(\beta_i/p_0)$. Let us denote by $P$ the discrete Preisach operator obtained with this scaling. Then equation (6) becomes

$$\Delta r_t + \kappa \Delta (Pr)_t = \eta_t \sqrt{h} + \left( a - \frac{1}{2} \right) h$$

(7)

where $\Delta (Pr)_t = (Pr)_{t+h} - (Pr)_t$. We will consider some simulations of this price dynamics model in Section 5. The formal continuous time counterpart of equation (7),

$$dr_t + \kappa d(Pr)_t = \left( a - \frac{1}{2} \right) dt + dW_t$$

(8)

has the form of equation (1) with constant coefficients.

A few remarks are in order. First, equation (8) can be integrated explicitly in terms of the inverse Preisach operator, namely equation (8) is equivalent to

$$r_t - r_0 = (I + \kappa P)^{-1} [(a - 1/2)(t - t_0) + W_t - W_0]$$

(9)

where $I$ is the identity operator and $t_0, r_0, W_0$ are the initial conditions (for details regarding the properties of the inverse operator $(I + \kappa P)^{-1}$, see [8]). This formula provides a rigorous interpretation of solutions to continuous time equation (8) and can be used to prove the convergence of the discretisations (7) to the continuous time limit (8) as $h \to 0$. A similar formula and the same approach can be used when the trend term $a = a(t)$ is a function of time (for example, accounting for seasonality in the trend). However, if $a$ depends on the price, then (8) becomes a closed loop system not amenable to the explicit integration. In this case, a mathematical formalism for the continuous time equation (8) is an open problem as it is for the more general equation (1).

As the second remark, equation (8), when rewritten in the form (1), has the parameter $\xi = 1/\kappa$. This quantity can be assumed to be of order 1. However, in models of economics
systems proposed in [5] (as well as in the hydrological context, see [3]) the cases of interest are $\xi \ll 1$ and $\xi = 0$. The case $\xi = 0$ can be singular and requires special treatment (as can be seen, for example, from equations (11), (10) below where $l_n = 0$ at every turning point of a solution to the discrete time counterpart of (1)). For example, in the deterministic setting, it has been shown that solutions of equation (1) with $b = 0$ have better regularity for $\xi > 0$ than for $\xi = 0$. Moreover, non-uniqueness of solutions to the initial value problem is possible for $\xi = 0$, see [3]. Nevertheless, natural generic conditions ensure the uniqueness property and the convergence of the solution of the deterministic equation with $\xi > 0$ to the solution of the equation with $\xi = 0$ as $\xi \to 0$. Numerical schemes for solving deterministic equations with $\xi = 0$ have been developed in [24–26]; the limit $\xi \to 0$ have been used in [16] to solve equations with a stochastic component that have piecewise smooth trajectories.

4. Numerical schemes

In this section we outline two Euler type time discretisations of equation (1). The two schemes differ in the method we use to evaluate the change in the output of the Preisach operator on each time step. The first scheme we call the rectangular method. Here the increment $\Delta y_n$ of the output is evaluated in the same way as proposed in the deterministic setting in [24–26]. It is assumed that the time step is sufficiently small so that (i) the shaded area $S(t_{n+1}) \setminus S(t_n)$ shown in figure 5 (a) is well approximated by the rectangle which has the width equal to the increment $\Delta x_n$ of the input and the length equal to the last link of the staircase line $S(t_n)$; (ii) a typical $\Delta x_n$ is much smaller than $\xi$ and hence the increment $\Delta y_n$ of the output at the turning points of $x_n$ is close to $\xi \Delta x_n$; and, (iii) for most other (non-turning) points of $x_n$ the set $S(t_{n+1}) \setminus S(t_n)$ is trapezoid as in figure 5 (a).

Alternatively, we calculate $\Delta y_n$ using the exact expression for the shaded area shown in figure 5 and call the corresponding numerical scheme the triangular method.

To simplify the notation we consider the Preisach operator which has the constant weight function $\mu(\alpha, \beta) = 1$ in the area of variation of $x_n$.

4.1. Rectangular method

We replace the right hand side of equation (1) with the finite difference

$$d_n = a(x_n, t_0 + nh)h + b(x_n, t_0 + nh)\sqrt{n}\Delta W_n$$

where $h$ is the time step and $\Delta W_n$ are independent Gaussian random variables

$$\Delta W_n \sim N(0, 1)$$

In the rectangular method, we approximate the increment of the output of the Preisach operator on the $n$-th time step by the product of the increment $\Delta x_n = x_{n+1} - x_n$ of the input and the length $l_n$ of the last link of the state of the Preisach operator at the moment $n$. This leads to the discretisation of equation (1) of the form

$$(l_n + \xi)\Delta x_n = d_n$$

where

$$l_n = \begin{cases} 
  x_n - m^n_0 & \text{if } d_n > 0, d_{n-1} > 0 \\
  0 & \text{if } d_n d_{n-1} < 0 \\
  M^n_0 - x_n & \text{if } d_n < 0, d_{n-1} < 0 
\end{cases}$$

(11)

Here $m^n_0$ and $M^n_0$ are the first elements of the arrays $\omega_m$ and $\omega_M$ of the main minima and maxima at the moment $n$; $d_{-1}$ equals the value $\zeta_0$ of the variable $\zeta$ at the initial moment. We
note that the sign of the increment $\Delta x_n$ coincides with the sign of $d_n$; $l_n$ is zero at the turning points of $x_n$, that is $l_n = 0$ whenever $\Delta x_n \Delta x_{n-1} < 0$ (equivalently, $d_n d_{n-1} < 0$), while $l_n > 0$ whenever $\Delta x_n \Delta x_{n-1} > 0$ (equivalently, $d_n d_{n-1} > 0$).

The numerical scheme has been implemented in the following steps to account for updating the memory arrays $\omega_m$ and $\omega_M$, the variable $\zeta_n$ and the value of $x_n$. For $t = t_0 + nh$:

(i) Calculate the value of $d_n$;

(ii) Evaluate for turning points as follows:

\begin{itemize}
  \item If $d_n d_{n-1} < 0$ (i.e., $x_n$ is a turning point), then:
    \begin{itemize}
      \item If $d_n > 0$ add the element $x_n$ to the memory array $\omega_m$ and set $\zeta_{n+1} = 1$;
      \item If $d_n < 0$ add the element $x_n$ to the array $\omega_M$ and set $\zeta_{n+1} = -1$;
    \end{itemize}
  \item If $d_n d_{n-1} > 0$ then the memory arrays $\omega_m$, $\omega_M$ remain the same and $\zeta_{n+1} = \zeta_n$;
\end{itemize}

(iii) Calculate the new $x$ value as $x_{n+1} = x_n + \frac{d_n}{l_n}$;

(iv) Update the memory arrays for the new position of $x$ by performing deletions of their elements if either of the following conditions are met:

\begin{itemize}
  \item For $d_n > 0$ remove all the elements $M_0^n, \ldots, M_k^n$ satisfying $M_i^n \leq x_{n+1}$ from the array $\omega_M$; remove the same number of elements $m_0^n, \ldots, m_k^n$ from the array $\omega_m$;
  \item For $d_n < 0$ remove all the elements $m_0^n, \ldots, m_k^n$ satisfying $m_i^n \geq x_{n+1}$ from the array $\omega_m$ and the same number of elements $M_0^n, \ldots, M_k^n$ from $\omega_M$.
\end{itemize}

4.2. Triangular method

In this alternative method, we use the exact expression for the increment $\Delta y_n$ of the output of the Preisach operator corresponding to the increment $\Delta x_n$ of the input, i.e.,

$$\Delta y_n = (Px)(t_0 + (n + 1)h) - (Px)(t_0 + nh)$$

where $x(t)$ is a monotone interpolation of the input between the discrete values $x_n = x(t_0 + nh)$ and $x_{n+1} = x(t_0 + (n + 1)h)$. This leads to the discretisation

$$\xi \Delta x_n + \Delta y_n = d_n \quad (12)$$

of equation (1) with the same $d_n$ as above. This formula defines $x_{n+1}$ implicitly given the value $x_n$ and the state of the Preisach operator (i.e., the arrays $\omega_m$ and $\omega_M$ and the variable $\zeta$) at the moment $n$; again, the sign of the increment $\Delta x_n$ equals the sign of $d_n$ and $\Delta y_n$ has the same sign. Solving equation (12) for $x_{n+1}$ is equivalent to explicit inversion of the operator $\xi I + P$ and leads to four cases depending on the signs of $d_n$ and $\zeta_n$. The geometrical interpretation considered in Section 2 makes this inversion straightforward (cf. Figure 5).

Let us consider in some detail the case where $d_n > 0$ and $\zeta_n = 1$, i.e., the last link of the state is horizontal at the moment $n$. An iterative procedure is used to find $x_{n+1}$ with the number of iterations depending on how many corners should be erased from the state of the Preisach operator when the input increases from $x_n$ to $x_{n+1}$. We introduce the value

$$\Sigma_0 = \xi (M_0^n - x_n) + \frac{(M_0^n - m_0^n)^2}{2} - \frac{(x_n - m_0^n)^2}{2}$$

If $\Sigma_0 \geq d_n$ then $x_{n+1} \leq M_0^n$ (i.e., the value of $x_{n+1}$ lies below the first corner of the staircase state $S(t_0 + nh)$) and equation (12) is equivalent to

$$\xi (x_{n+1} - x_n) + \frac{(x_{n+1} - m_0^n)^2}{2} - \frac{(x_n - m_0^n)^2}{2} = d_n$$
hence
\[ x_{n+1} = m_0^n - \xi + \sqrt{(m_0^n - \xi)^2 - ((m_0^n)^2 - (x_n - m_0^n)^2 - 2(\xi x_n + d_n))} \]
where the positive root is chosen to ensure that \( x_{n+1} > x_n \). If however \( \Sigma_0 < d_n \) then we compare \( d_n \) consecutively with the elements \( \Sigma_1, \Sigma_2, \ldots \) of the sequence
\[ \Sigma_k = \Sigma_{k-1} + \frac{(M_k^n - m_k^n)^2}{2} - \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} + \xi(M_k^n - M_{k-1}^n) \]
until a \( \Sigma_k \) is found such that \( \Sigma_k \geq d_n > \Sigma_{k-1} \). These relations ensure that \( M_{k-1}^n < x_{n+1} \leq M_k^n \) and equation (12) is equivalent to
\[ \Sigma_{k-1} + \xi(x_{n+1} - M_{k-1}^n) + \frac{(x_n + m_k^n)^2}{2} - \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} = d_n \]
which implies
\[ x_{n+1} = m_k^n - \xi + \sqrt{(m_k^n - \xi)^2 - ((m_k^n)^2 - (M_{k-1}^n - m_{k-1}^n)^2 - 2(\xi M_{k-1}^n + d_n - \Sigma_{k-1}))} \]
A similar algorithm for solving equation (12) applies in the three other cases. For \( d_n < 0 \), \( \zeta_n = 1 \), we set
\[ \Sigma_{-1} = 0; \quad \Sigma_k = \Sigma_{k-1} - \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} + \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} + \xi(m_k^n - m_{k-1}^n), \quad k = 0, 1, 2, \ldots \]
then relations \( \Sigma_k \leq d_n < \Sigma_{k-1} \) imply
\[ x_{n+1} = M_{k-1}^n + \xi - \sqrt{(M_{k-1}^n + \xi)^2 - ((M_{k-1}^n)^2 - (M_{k-1}^n - m_{k-1}^n)^2 - 2(\xi m_{k-1}^n + d_n - \Sigma_{k-1}))} \]
where we use the convention \( M_{-1}^n = m_{-1}^n = x_n \). For \( d_n > 0 \), \( \zeta_n = -1 \), the relations \( \Sigma_{k+1} < d_n \leq \Sigma_k \) with
\[ \Sigma_{-1} = 0; \quad \Sigma_k = \Sigma_{k-1} + \frac{(M_k^n - m_k^n)^2}{2} - \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} + \xi(M_k^n - M_{k-1}^n), \quad k = 0, 1, 2, \ldots \]
imply
\[ x_{n+1} = m_{k-1}^n - \xi + \sqrt{(m_{k-1}^n - \xi)^2 - ((m_{k-1}^n)^2 - (M_{k-1}^n - m_{k-1}^n)^2 - 2(\xi M_{k-1}^n + d_n - \Sigma_{k-1}))} \]
For \( d_n < 0 \), \( \zeta_n = -1 \), the relations \( \Sigma_k \leq d_n < \Sigma_{k-1} \) with
\[ \Sigma_{-1} = 0; \quad \Sigma_k = \Sigma_{k-1} - \frac{(M_k^n - m_k^n)^2}{2} + \frac{(M_{k-1}^n - m_{k-1}^n)^2}{2} + \xi(m_k^n - m_{k-1}^n), \quad k = 0, 1, 2, \ldots \]
imply
\[ x_{n+1} = M_k^n + \xi - \sqrt{(M_k^n + \xi)^2 - ((M_k^n)^2 - (M_{k-1}^n - m_{k-1}^n)^2 - 2(\xi m_{k-1}^n + d_n - \Sigma_{k-1}))} \]
The implementation of the triangular method has the same steps (i), (ii) and (iv) as described in the previous subsection for the rectangular method. Step (iii) is modified and \( x_n \) is updated to obtain the next iteration \( x_{n+1} \) according to the algorithm presented in this subsection.
Table 1.

| ξ  | h   | x₀   |
|----|-----|------|
| 0.1| 0.01| 2.0  |

5. Results

As the first model example, we consider discretisations of the formal equation

$$\xi dx_t + d(Px)_t = -x_t dt + dW_t$$

obtained by the rectangular and triangular methods using different time step size $h$. We compare trajectories obtained by the two methods for the same realisation of the discretised Wiener process (random walk) $W_t$. We also compare the trajectories of equation (13) with the trajectories of the Ornstein-Uhlenbeck equation

$$\xi dx_t = -x_t dt + dW_t$$

obtained by the Euler method.

As the second example, the rectangular and triangular discretisations of the price dynamics model (8) are considered.

In all the examples, the Preisach weight function is set to $\mu(\alpha, \beta) \equiv 1$ for the whole domain of the Preisach plane where relays experience at least one switch during the simulation. The initial state of the Preisach operator is the staircase with all the steps of size 1.

5.1. Rectangular method

The values of parameters used in the simulations shown in this subsection are given in table 1. A numerical realisation of the rectangular method for equation (13) and the Euler method for equation (14) using these parameters is shown in figure 7; $x_0$ is the initial value of the solution.

![Graphs](image-url)

**Figure 7.** Data generated from the implementation of the rectangular method. (a) The red line is the trace of solution $x_t$ to equation (13) including the Preisach operator. The black line denotes the solution of equation (14) without the Preisach operator for the same realisation of $W_t$. (b) A zoom of the region indicated in (a).

A look at figure 7 (b) which presents a zoomed in part of the full trajectory shows that while solutions of discretisations of equations (13) and (14) experience ups and downs at roughly
the same time moments for the same realisation of the random walk, the inclusion of the Preisach operator has limited the effect of the jumps. It appears that the introduction of the Preisach operator in the stochastic equation (14) acts as a damping factor to the trajectory. It is characteristic of the damping that it is stronger on the parts of the trajectory demonstrating a stronger upwards or downwards trend. This feature can be explained by formulas (11) and (10). If the solution satisfies $x_n \geq x_{n_0}$ on some time interval $n_0 \leq n \leq N$, then at every moment $n$ from this interval such that $x_n > x_k$ for all $n_0 \leq k \leq n$, the quantity $l_n$ also satisfies $l_n > l_k$ for all $n_0 \leq k \leq n$. In this sense, the upward trend of the solution leads to the upward trend of $l_n$ (and, similarly, the downward trend of $x_n$ leads to the upward trend of $l_n$). At the same time, relation (10) implies stronger damping of the increment $\Delta x_n = x_{n+1} - x_n$ for larger $l_n$.

5.2. Triangular method
The parameters taken for the implementation of the triangular method for equation (13) and the Euler method for equation (14) are shown in table 2. Figure 8 presents solutions obtained with these parameters for the same realisation of $W_t$. The Preisach operator term seems to be a limiting factor on the size of jumps, as is the case with the implementation of the rectangular method. Figure 8b shows that while the two trajectories undergo up and down movements at the same time the inclusion of the Preisach operator has caused the changes to be less severe.

![Figure 8](image-url)  

(a) Data generated from the implementation of the triangular method for equation (13) (red) and the Euler method for equation (14) (black), (b) A zoom of the region indicated in (a).

5.3. Comparison of triangular and rectangular methods
Figures 9 and 10 show a realisation of the rectangular and triangular methods for equation (13). The parameters used for the generation of these results are $\xi = 0.1$ and $x_0 = 2$; the value of $h$ for each plot is stated in the caption. We see that the agreement between the two methods improves for smaller time step $h$. In particular, for the given parameters, a good agreement...
between the two methods is achieved at $h = 0.00001$, see figure 10. Here the trajectories are close apart from a few discrepancies at some turning points where the rectangular method tends to produce larger jumps for the chosen small value of the parameter $\xi$.

**Figure 9.** Solution of the discretisations of equation (13) as generated from the implementation of both the rectangular method (black) and the triangular method (red) for different time step size: (a) $h = 0.1$, (b) $h = 0.001$.

**Figure 10.** Same as Figure 9 for $h = 0.00001$, (b) A zoom of the region marked in (a).

5.4. Price dynamics model
We now consider the price dynamics model (8) presented in Section 3. Figure 11 compares the results of implementation of each of the rectangular and triangular numerical methods in solving equation (8) with positive $\kappa$ with the solution of the same equation for $\kappa = 0$ (i.e., equation...
Figure 11. The red line shows the plot of a solution $r(t)$ of equation (8) as generated from the implementation of (a) the rectangular method, and (b) the triangular method. Parameters are presented in table 3. The black line in each panel is the solution of equation (8) with $\kappa = 0$, which coincides with the realisation of the random walk $W_t$ used in the red plot for the parameter values from the table.

$$dr_t = (a - 1/2)dt + dW_t$$ (without the Preisach operator term) obtained by the Euler method.

The values of the parameters used for these plots are shown in table 3. The plots demonstrate the same features as we observed in the other examples above. We see that solutions of the model including the Preisach operator ($\kappa = 1$) and the model without the Preisach operator ($\kappa = 0$) have similar shape, however the inclusion of the Preisach operator leads to a damping effect. In the model including the Preisach operator the price does not go as high, as well as falls in the price are limited and less severe, as in the model with $\kappa = 0$.

6. Conclusions
We have proposed and tested two discrete time counterparts of the formal continuous time stochastic differential equation (1) involving the Preisach operator. These two numerical schemes have been applied to an Ornstein-Uhlenbeck type equation and to a simple price dynamics model involving the Preisach memory term. We have shown that there is a good agreement between the two methods once the time step is sufficiently small. It has also been shown that the inclusion of the Preisach operator acts as a limiting damping effect when compared with an equivalent form of a regular stochastic differential equation. The damping is more pronounced on the parts of the trajectory demonstrating a stronger upwards or downwards trend. This effect can be explained by the fact that both upwards or downwards trend of the trajectory result in an upwards trend of the quantity (11). The results of this paper are preliminary to a broader programme of research of stochastic equations with the Preisach operator outlined in the Introduction.

Acknowledgments
The authors are thankful to Prof. Harbir Lamba for valuable discussions of the results of the paper and to Prof. Mihai Dimian and Prof. Ram Iyer for instructive discussions of numerical algorithms for solving discrete and stochastic equations with the Preisach operator. The work was partially supported by Russian Foundation for Basic Research, grant 10-01-93112.

References
[1] O’Kane J P 2005 J. Phys.: Conf. Ser. 22 148
[2] Flynn D, McNamara H, O’Kane J P and Pokrovskii A 2006 The Science of Hysteresis vol III, ed I Mayergoyz and G Bertotti (Elsevier, Academic Press) 689
[3] Appelbe B, Flynn D, McNamara H, O’Kane P, Pimenov A, Pokrovskii A, Rachinskii D and Zhezherun A 2009 *IEEE Control Systems Magazine* **29** 44

[4] Cross R, Grinfeld M and Lamba H 2009 *IEEE Control Systems Magazine* **29** 30

[5] Cross R, McNamara H and Pokrovskii A V 2008 *Physica B* **403** 451

[6] Mayergoyz I D 2006 *The science of hysteresis* vol I, ed I Mayergoyz and G Bertotti (Elsevier, Academic Press) 529

[7] Krasnosel’skii M A and Pokrovskii A V 1989 *Systems with hysteresis* (Springer)

[8] Brokate M and Sprekels J 1996 *Hysteresis and phase transitions* (Springer)

[9] Mayergoyz I D and Korman C E 1991 *J. Appl. Phys.* **69** 2128

[10] Rugkwamsook P, Korman C, Bertotti G and Pasquale M 1999 *J. Appl. Phys.* **85** 4361

[11] Mayergoyz I D, Adly A, Huang M W and Kraft C 2000 *IEEE Trans. Magn.* **36** 3208

[12] Dimian M, Coca E and Popa V 2009 *J. Appl. Phys.* **105** 07D515

[13] Dimian M, Adedoyin A, Gindulescu A and Andrei P 2009 *IEEE Trans. Magn.* **45** 5231

[14] Spanos P D, Cacciola P and Redhors J 2004 *Nonlinear Dynamics* **36** 405

[15] Rychlik I and Gupta S 2007 *Int. J. of Fatigue* **29** 406

[16] Brokate M, McCarthy S, Pimenov A, Pokrovskii A and Rachinskii D 2010 *Environ. Model. Assess.* in press

[17] Lamba H and Seaman T 2008 *Int. J. of Theoretical and Applied Finance* **11** 717

[18] Cross R, Grinfeld M, Lamba H and Seaman T 2007 *The European Phys. Journal B* **57** 213

[19] Cross R, Grinfeld M, Lamba H and Seaman T 2005 *Physica A* **354** 463

[20] Lamba H and Seaman T 2009 arXiv:0709.4242, *Physica A* in press

[21] Lamba H 2009 arXiv:0908.0949v1

[22] Appleyby J A D and Wu H 2008 *J. Phys.: Conf. Ser.* **138** 012002

[23] Mayergoyz I 2003 *Mathematical models of hysteresis and their applications* (New York: Elsvier)

[24] Flynn D and Rasskazov O 2005 *J. Phys.: Conf. Ser.* **22** 43

[25] Flynn D, O’Kane J P and Zhezherun A 2006 *J. Phys.: Conf. Ser.* **55** 63

[26] Flynn D, McNamara H, Pokrovskii A and O’Kane J P 2008 *Physica B* **403** 440