"Exact" numerical algorithms for linear stochastic wave equation and stochastic Klein-Gordon equation [1]

A. Martin[2], S.M. Prigarin[3] and G. Winkler[4]

On the basis of integral representations we propose numerical methods to solve the stochastic wave equation and the stochastic Klein-Gordon equation. The algorithms are exact in a probabilistic sense.

1 The Cauchy problem for stochastic wave equation. Integral representation

Let us consider the Cauchy problem for the stochastic wave equation

$$\frac{\partial^2 u}{\partial t^2}(t,x) - a^2 \frac{\partial^2 u}{\partial x^2}(t,x) = g(t,x,u(t,x)) + f(t,x,u(t,x))\, w(t,x), \tag{1}$$

$$u(0,x) = u_0(x), \quad \frac{\partial u}{\partial t}(0,x) = \tilde{u}_0(x), \tag{2}$$

where $a > 0$, $x \in R$, $t \in R^+$, and w is a Gaussian white noise on the plane.

The aim is to numerically solve the Cauchy problem for the stochastic wave equation, i.e., to simulate realizations of the random field $u(t,x)$. To this end we need an exact definition of the solution of problem (1), (2). The corresponding formalism can be found, for example, in the papers [1], [2], [3]. For our purposes it is enough to assume that the solution $u(t,x)$ of problem (1), (2) can be defined as a unique solution to the stochastic integral equation

$$u(t,x) = i(t,x) + \int_{R_+} \int_R G(t,x,s,y) g(s,y,u(s,y))\, dy ds + \tag{3}$$

$$+ \int_{R_+} \int_R G(t,x,s,y) f(s,y,u(s,y))\, W(ds\, dy).$$

Here

$$i(t,x) = \frac{1}{2}\left[u_0(x+at) + u_0(x-at)\right] + \frac{1}{2a} \int_{x-at}^{x+at} \tilde{u}_0(y) dy,$$

[1]The work partially was supported by RFBR grant N 00-15-96173

[2]Institute of Biomathematics and Biometry, GSF-National Research Center for Environment and Health, 85764 Neuherberg München, Germany; e-mail: martin@gsf.de

[3]Institute of Computational Mathematics and Mathematical Geophysics, pr. Lavrentieva 6, Novosibirsk, 630090, Russia; e-mail: smp@osmf.sscc.ru

[4]Institute of Biomathematics and Biometry, GSF-National Research Center for Environment and Health, 85764 Neuherberg München, Germany; e-mail: gwinkler@gsf.de

G is Green's function for the wave equation,

$$G(t,x,s,y) = 1_{\Delta(t,x)}(s,y)/(2a),$$

$$\Delta(t,x) = \{(s,y) \in R_+ \times R : 0 \le s \le t, |x-y| \le a(t-s)\}, \quad (t,x) \in R_+ \times R,$$

and W is a Gaussian stochastic measure with the properties

$$EW(A) = 0, \quad EW(A)W(B) = m(A \cap B),$$

where A, B are measurable sets and m is the Lebesgue measure on the plane.

In particular, a solution u of the integral equation (3) exists (and it is unique and almost surely continuous) if the following conditions are fulfilled [2, 3]:

(A) there exist strictly positive constants $C(g)$ and $C(f)$ such that

$$|g(t,x,y) - g(t,x,z)| \le C(g)|y-z|, \quad |f(t,x,y) - f(t,x,z)| \le C(f)|y-z|$$

for all $t \in R_+$ and x, y, $z \in R$;

(B) for all $(t,x) \in R_+ \times R$ the function $(s,y) \mapsto G(t,x,s,y)g(s,y,0)$ is integrable and the function $(s,y) \mapsto G(t,x,s,y)f(s,y,0)$ is square integrable,

(C) the functions u_0, $\tilde{u}_0$ are continuous and $\int_R |u_0(x)|^2\,dx + \int_R |\tilde{u}_0(x)|^2\,dx < \infty$.

If the functions g and f do not depend on u, then the solution is given by

$$u(t,x) = i(t,x) + \int_{R_+}\int_R G(t,x,s,y)g(s,y)\,dyds + \tag{4}$$

$$+ \int_{R_+}\int_R G(t,x,s,y)f(s,y)\,W(ds\,dy).$$

In this case the random function u is Gaussian, and an "exact" numerical algorithm can be constructed on the basis of representation (4).

2 "Exact" numerical algorithms for g and f independent of u

2.1 Algorithm for the Cauchy problem without boundaries

A realization of the solution u for the Cauchy problem (1), (2) will be simulated on a grid G_u,

$$G_u = \{(t_j, x_k) = (jh, akh)\},$$

$$j \in \{0, \frac{1}{2}, 1, 1\frac{1}{2}, 2, 2\frac{1}{2}, \ldots\}, \quad k \in \{\ldots, -1\frac{1}{2}, -1, -\frac{1}{2}, 0, \frac{1}{2}, 1, \ldots\},$$

where h is a time step, ah is a spatial step and for a grid point the indices j, k are simultaneously either integers or fractional numbers.

In addition to the values $u_k^j = u(jh, akh)$ on the grid G_u the algorithm operates with the values

$$v_k^j = \int_{x_k - ah/2}^{x_k + ah/2} \frac{\partial u}{\partial t}(t_j, y)dy$$

on an adjacent grid G_v

$$G_v = \{(t_j, x_k) = (jh, akh)\}$$

with the same sets for the indices j, k. The difference from the grid G_u is that for any point of G_v one of the indices is an integer while another one is fractional.

The algorithm can be described by the following steps.

Initialization. We set $j = 0$ and from the initial conditions (2) we find the values u_k^0, $v_{k+1/2}^0$ for $k \in \{\ldots, -1, 0, 1, \ldots\}$.

Step 1. Equation (4) gives an explicit rule to calculate $u^{j+1/2}$:

$$u_{k+1/2}^{j+1/2} = \left(\frac{1}{2}\left(u_k^j + u_{k+1}^j\right) + \frac{1}{2a} v_{k+1/2}^j\right) + \xi_{k+1/2}^{j+1/2}, \tag{5}$$

where

$$\begin{aligned} \xi_{k+1/2}^{j+1/2} &= \int_{t_j}^{t_{j+1/2}} \int_{x_k}^{x_{k+1}} G(t_{j+1/2}, x_{k+1/2}, s, y) g(s,y)\, dy\, ds \\ &+ \int_{t_j}^{t_{j+1/2}} \int_{x_k}^{x_{k+1}} G(t_{j+1/2}, x_{k+1/2}, s, y) f(s,y)\, W(ds\, dy) \end{aligned}$$

is a Gaussian random variable with variance

$$\int_{t_j}^{t_{j+1/2}} \int_{x_k}^{x_{k+1}} G^2(t_{j+1/2}, x_{k+1/2}, s, y) f^2(s,y)\, dy\, ds.$$

Note, that to simulate the values $u_{k+1/2}^{j+1/2}$ for the time level $j + 1/2$ we use only the values u_k^j, $v_{k+1/2}^j$ from the time level j and independent random variables $\xi_{k+1/2}^{j+1/2}$.

Step 2. Now we want to calculate the values of $v_{k+1}^{j+1/2}$. To obtain an expression for $v_{k+1}^{j+1/2}$ we use two different representations for u_{k+1}^{j+1}, namely

$$u_{k+1}^{j+1} = \left(\frac{1}{2}\left(u_{k+1/2}^{j+1/2} + u_{k+3/2}^{j+1/2}\right) + \frac{1}{2a} v_{k+1}^{j+1/2}\right) + \xi_{k+1}^{j+1}, \tag{6}$$

$$\begin{aligned} u_{k+1}^{j+1} = &\left(\frac{1}{2}\left(u_k^j + u_{k+2}^j\right) + \frac{1}{2a}\left(v_{k+1/2}^j + v_{k+3/2}^j\right)\right) + \\ &+ \left(\xi_{k+1/2}^{j+1/2} + \xi_{k+3/2}^{j+1/2} + \delta_{k+1}^j + \xi_{k+1}^{j+1}\right), \end{aligned} \tag{7}$$

where

$$\begin{aligned} \delta_{k+1}^j &= \frac{1}{2a} \int_{t_j}^{t_{j+1/2}} \int_{x_{k+1}-a(s-t_j)}^{x_{k+1}+a(s-t_j)} g(s,y)\, dy\, ds \\ &+ \frac{1}{2a} \int_{t_j}^{t_{j+1/2}} \int_{x_{k+1}-a(s-t_j)}^{x_{k+1}+a(s-t_j)} f(s,y)\, W(ds\, dy) \end{aligned}$$

is a Gaussian random variable with variance

$$\left(\frac{1}{2a}\right)^2 \int_{t_j}^{t_{j+1/2}} \int_{x_{k+1}-a(s-t_j)}^{x_{k+1}+a(s-t_j)} f^2(s,y)\, dy\, ds.$$

A combination of (6) and (7) yields

$$v_{k+1}^{j+1/2} = 2a\left[\frac{1}{2}\left(u_k^j + u_{k+2}^j\right) + \frac{1}{2a}\left(v_{k+1/2}^j + v_{k+3/2}^j\right) + \right. \tag{8}$$

$$+\left(\xi_{k+1/2}^{j+1/2}+\xi_{k+3/2}^{j+1/2}\right)+\delta_{k+1}^{j}-\frac{1}{2}\left(u_{k+1/2}^{j+1/2}+u_{k+3/2}^{j+1/2}\right)\Big].$$

Thus, at this step the values $v_{k+1}^{j+1/2}$ are simulated according to expression (8). Note, that all the components of the expression, except δ_{k+1}^{j}, are defined at the previous steps of the algorithm, and the random variables ξ_{k+1}^{j+1} appear in (6) and (7) but cancel out in (8).

Cycling. Set $j = j + 1/2$ and go back to step 1.

2.2 Algorithm for the Cauchy problem with the Dirichlet boundary conditions

Assume now that the random function $u(t,x)$, $x \in (A,B)$, $t \in R^+$, is a solution of the Cauchy problem (1), (2) with the Dirichlet boundary conditions

$$u(t,A) = u(t,B) = 0. \tag{9}$$

For the numerical modelling of the Gaussian field $u(t,x)$ we should modify the previous algorithm. To simplify the notation, hereafter we set $A = 0$, $B = 1$.

Green's function for problem (1), (2), (9) can be written down in the form

$$\begin{aligned}\Sigma(t,x,s,y) &= \sum_k \Big[G(t,x+2k;s,y)1_{[0,1]}(y)\\ &\qquad -G(t,-x+2k;s,y)1_{[0,1]}(y)\Big].\end{aligned}$$

The solution of the Cauchy problem with the Dirichlet boundary conditions is given by the following equation (cf.[2])

$$u(t,x) = i_D(t,x) + \int_{R_+}\int_0^1 \Sigma(t,x,s,y)g(s,y)\,dy\,ds+ \tag{10}$$

$$+\int_{R_+}\int_0^1 \Sigma(t,x,s,y)f(s,y)\,W(ds\,dy),$$

where

$$\begin{aligned}i_D(t,x) = \frac{1}{2}\sum_k &\Big[u_0(x+2k-at)1_{(0,1)}(x+2k-at)\\ + &\ u_0(x+2k+at)1_{(0,1)}(x+2k+at)\\ - &\ u_0(-x+2k-at)1_{(0,1)}(-x+2k-at)\\ - &\ u_0(-x+2k+at)1_{(0,1)}(-x+2k+at)\Big]\\ +\frac{1}{2a}\sum_k &\Big[\int_{(x+2k-at)\vee 0}^{(x+2k+at)\wedge 1}\tilde{u}_0(y)dy - \int_{-x+2k-at)\vee 0}^{-x+2k+at)\wedge 1}\tilde{u}_0(y)dy\Big].\end{aligned}$$

Representation (10) implies the following modification of the numerical algorithm. For the step size ah in space, we assume that $1/ah = N$ is an integer.

Initialization. Set $j = 0$ and determine u_k^0 for $k \in \{0,\ldots,N\}$ and $v_{k+1/2}^0$ for $k \in \{0,\ldots,N-1\}$ according to the initial conditions (2).

Step 1. The values $u_{k+1/2}^{j+1/2}$ for $k \in \{0,\ldots,N-1\}$ are calculated by (5).

Step 2. The values $v_{k+1}^{j+1/2}$ for $k \in \{0, \ldots, N-2\}$ are calculated by (8).

Step 3. The values u_k^{j+1} for $k \in \{1, \ldots, N-1\}$ are calculated according to (5), while u_0^{j+1} and u_N^{j+1} are defined by the boundary conditions, i.e., $u_0^{j+1} = u_N^{j+1} = 0$.

Step 4. Calculation of v_{k+1}^{j+1}. For $k \in \{\frac{1}{2}, \frac{3}{2}, \ldots, \frac{N-5}{2}\}$ we use (8). To find $v_{1/2}^{j+1}$ and $v_{N-1/2}^{j+1}$ at the boundaries we should apply the boundary conditions. In particular, we have (note, that $u_0^{j+1} = 0$)

$$\begin{aligned} u_{1/2}^{j+3/2} &= \left(\frac{1}{2}\left(u_0^{j+1} + u_1^{j+1}\right) + \frac{1}{2a}v_{1/2}^{j+1}\right) + \xi_{1/2}^{j+3/2} \\ &= \left(\frac{1}{2}u_1^{j+1} + \frac{1}{2a}v_{1/2}^{j+1}\right) + \xi_{1/2}^{j+3/2}, \end{aligned}$$

$$\begin{aligned} u_{1/2}^{j+3/2} &= \left(\frac{1}{2}\left(-u_{1/2}^{j+1/2} + u_{3/2}^{j+1/2}\right) + \frac{1}{2a}v_1^{j+1/2}\right) \\ &+ \left(\xi_1^{j+1} + \delta_{1/2}^{j+1/2} + \xi_{1/2}^{j+3/2}\right). \end{aligned}$$

From these two representations we obtain:

$$\begin{aligned} v_{1/2}^{j+1} &= 2a\left[\frac{1}{2}\left(-u_{1/2}^{j+1/2} + u_{3/2}^{j+1/2}\right) + \frac{1}{2a}\left(v_1^{j+1/2}\right)\right. \\ &\left. + \left(\xi_1^{j+1}\right) + \delta_{1/2}^{j+1/2} - \frac{1}{2}\left(u_1^{j+1}\right)\right]. \end{aligned}$$

Similarly, for the other boundary we can obtain

$$\begin{aligned} v_{N-1/2}^{j+1} &= 2a\left[\frac{1}{2}\left(u_{N-3/2}^{j+1/2} - u_{N-1/2}^{j+1/2}\right) + \frac{1}{2a}\left(v_{N-1}^{j+1/2}\right)\right. \\ &\left. + \left(\xi_{N-1}^{j+1}\right) + \delta_{N-1/2}^{j+1/2} - \frac{1}{2}\left(u_{N-1}^{j+1}\right)\right]. \end{aligned}$$

Cycling. Set $j = j + 1$ and go back to Step 1.

Remark. The two algorithms are "exact" in the sense that the finite-dimensional distributions of the field $u(t, x)$ on the grid G_u coincide with the finite-dimensional distributions of the numerical solution u_k^j. On the other hand, the algorithms are not stable (as well as the solution $u(t, x)$): a disturbance occurring in computation will spread along the characteristics of the equation without decreasing in time. The Markov property (the values for the next time level depend only on the values obtained at the previous time level) makes algorithms fairly attractive for the numerical simulation.

3 On "exact" algorithms for the stochastic Klein-Gordon equation

A solution for the Cauchy problem (1), (2) for the stochastic Klein-Gordon equation

$$\frac{\partial^2 u}{\partial t^2}(t, x) - \frac{\partial^2 u}{\partial x^2}(t, x) = \alpha u(t, x) + w(t, x)$$

can be written down in the following way [3]

$$u(t,x) = i_{KG}(t,x) + \int_{R_+} \int_R G_{KG}(t,x,s,y)\, W(ds\, dy), \tag{11}$$

where

$$i_{KG}(t,x) = \frac{\partial}{\partial t} \int_{x-t}^{x+t} G_{KG}(t,x,0,y) u_0(y)\, dy + \int_{x-t}^{x+t} G_{KG}(t,x,0,y) \tilde{u}_0(dy),$$

$$G_{KG}(t,x,s,y) := G(t,x,s,y) I_0 \left(\sqrt{\alpha((t-s)^2 - (x-y)^2)} \right),\ \alpha \geq 0,$$

$$G_{KG}(t,x,s,y) := G(t,x,s,y) J_0 \left(\sqrt{|\alpha|((t-s)^2 - (x-y)^2)} \right),\ \alpha < 0,$$

and I_0, J_0 are the Bessel functions.

An "exact" numerical algorithm can be constructed on the basis of representation (11). But now the situation is rather different in comparison with the algorithms considered in Sections 2.1 and 2.2:

- The Bessel functions in the integral representation make computations much more laborious.
- The algorithm does not have the Markov property anymore, and the Gaussian random variables that have to be simulated, are dependent in a complicated way. This means that, practically, the algorithm results in a general simulation of a Gaussian random vector with correlated components, and for a fine grid (when the dimension of the vector becomes too large) the algorithm may not be feasible.

In our opinion, more efficient numerical algorithms for the stochastic Klein-Gordon equation, as well as for the general stochastic wave equation (1), can be developed on the basis of an iterative method for the integral representation (3).

References

[1] J.B. Walsh. *An Introduction to Stochastic Partial Differential Equations.* In: R. Carmona, H. Kesten, J.B. Walsh (eds.). Ecole d'Eté de Probabilités de Saint Flour XIV. Lecture Notes in Mathematics 1180, pages 266-437. Berlin, Heidelberg, New York: Springer, 1986

[2] R.A. Carmona, D. Nualart. *Random Non-Linear Wave Equations: Smoothness of the Solutions.* Probability Theory and Related Fields 79. Springer-Verlag, 1988, pp. 469-508

[3] A. Martin. Hyperbolic Stochastic Partial Differential Equations: Small Balls and Simulation; Propagation of Singularities (Muenchen, Univ., Diss.). Aachen: Shaker, 2002