INTERNATIONAL JOURNAL OF NUMERICAL MODELLING: ELECTRONIC NETWORKS, DEVICES AND FIELDS
Int. J. Numer. Model. 2001; **14**:439–449 (DOI: 10.1002/jnm.425)

TLM-based solutions of the Klein–Gordon equation (Part I)

William J. O'Connor*,† and Fergus J. Clune

213 Mechanical Engineering, University College Dublin, Belfield, Dublin 4, Ireland

SUMMARY

The transmission line matrix (TLM) method has become well established as a numerical solution scheme for wave problems in electromagnetics and, to a lesser extent, in acoustics and mechanics. It has also been applied to diffusion/heat-conduction problems. Here the technique is extended to solving the Klein–Gordon equation that arises in Quantum Mechanics and in the dynamics of an elastically anchored vibrating string. In Part I, two novel, TLM-based algorithms are presented and verified. By considering them as solving a special case of the more general 'forced' wave equation, they illustrate how, with care, the TLM algorithm can be adapted to model a wide range of effects. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: TLM; transmission line matrix; dispersive wave equation; dispersion; Klein–Gordon equation; numerical methods

1. INTRODUCTION

The transmission line matrix (TLM) method in its basic formulation is a time-domain numerical solution technique for the wave equation. The method has been applied widely to electromagnetic problems and to a lesser extent to other wave phenomena, especially acoustics [1–3].‡

TLM methods can be extended beyond the basic wave equation to model lossy problems, where the defining differential equation, the telegrapher's equation, has an extra diffusion-type term corresponding to the loss effect. The TLM solution of the telegrapher's equation is well established. As the loss term becomes more significant, diffusion-type effects soon dominate, allowing a wide range of diffusion-type problems to be solved using TLM techniques [4,5].

In this paper, another extension of TLM techniques beyond the solution of the standard wave equation is presented. The form of the defining differential equation is

$$u_{tt} = c^2 u_{xx} - hu \tag{1}$$

* Correspondence to: Dr. William J. O'Connor, 213 Mechanical Engineering, University College Dublin, Belfield, Dublin 4, Ireland.
† E-mail: william.oconnor@ucd.ie

‡ Many electromagnetic applications of TLM are found throughout the following two journals: *IEEE Transaction on Microwave Theory and Techniques*; and in *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields.*

Published online 15 May 2001

Received 25 November 1998
Revised 8 March 1999

It arises in classical mechanics, where u represents, for example, the lateral displacement of a vibrating string under tension, but now with an extra, elastic acceleration or force (force proportional to the displacement) acting at every point. The typical case is that of a flexible string under tension embedded in a thin, elastic sheet held in a rigid frame. The same equation also arises in the quantum mechanics of scalar mesons where it is known as the 'Klein–Gordon' equation, the name by which it is generally known. In this paper, however, the discussion will focus on the problem of the elastically anchored vibrating string, and the hu term in Equation (1) will be referred to as the 'elastic' acceleration, proportional to the displacement. One important effect of the extra term is to make the wave problem dispersive [6,7]. That is, waves of different frequency are transmitted at different speeds, or equivalently, the phase relationship between harmonics is altered as they propagate, leading to distortion.

The challenge is to find a way to modify the basic TLM wave-propagation algorithm to model the extra physical effect properly. In total, four solution schemes are presented, two in Part I and two in Part II. All four methods are verified by comparing them (a) with corresponding finite-difference solution algorithms developed symbolically and (b) with analytical solutions derived by Fourier techniques.

2. TLM AND THE VIBRATING STRING

The partial differential equation for a vibrating string is perhaps the best-known example of the wave equation and probably the most frequently derived. If it is assumed that (a) the string is uniform with linear density (mass per unit length) ρ, (b) the tension T with vibration does not change locally from the static value, and (c) the displacement, u, from the stationary position is small at all points x along the string, (d) T/ρ is large with respect to the acceleration due to gravity, g and (e) the string is perfectly flexible, then the governing differential equation is easily shown to be

$$u_{tt} = (T/\rho)u_{xx} \tag{2}$$

The wave velocity, $c = \sqrt{(T/\rho)}$.

In solving wave problems by TLM, ideal impulses are assumed to travel in opposite directions in ideal transmission lines, each pulse travelling one matrix (or mesh) line, of length Δl, in one time interval Δt, before being scattered at mesh nodes. In the special case of one-dimensional waves (and therefore one-dimensional 'meshes') considered in this paper, the scattering becomes almost trivial: on scattering, pulses simply pass through each node to the next branch of the mesh. Also, in the one-dimensional case, the pulse velocity $\Delta l/\Delta t$ is identical to the unbounded wave velocity c. The generic TLM pulses in mesh link i will be referred to as r_i and l_i (for right-going and left-going, respectively): in the different methods in Part I and Part II these pulses will model different physical parameters, as will be explained.

To model a vibrating string by TLM the most obvious approach is to make these generic TLM pulses correspond to the components of the local string displacement, u. This is the approach taken in Part I. The actual string displacement at any point along the string is then the sum of the two pulses passing this point at this time. Thus, the two streams of TLM pulses become discretized versions of the well-known, counter-propagating, component solutions of the wave equation

$$u = f(x - ct) + g(x + ct) \tag{3}$$

with arbitrary functions f and g determined by initial and boundary conditions. Thus u, f, g, r_i and l_i all have dimensions of displacement (meters).

Boundary conditions for the string are easily modelled. For example, a fixed boundary means zero displacement, so that reflected wave pulses must be equal to the negative of the incident. Arbitrary initial conditions can be modelled by specifying—for the position—the value of the *sum* of the initial right- and left-going pulses and—for the velocity—the *gradients* (spatial derivatives) of these component functions. This second point will be clearer from the discussion below.

3. THE KLEIN–GORDON EQUATION

An extra, elastic force, $-(k\Delta x)\,u$, is now added to each element Δx of the string, where k is the elastic stiffness per metre length along the string, the force being proportional to the displacement u but tending to reduce it. The equation of motion from Newton's second law then becomes

$$Tu_{xx}\Delta x - k\Delta xu = \rho\Delta xu_{tt} \tag{4}$$

which, on rearrangement, becomes Equation (1) with $h = c^2k/T = k/\rho$, with dimension (s^{-2}).

To set up a TLM model of this equation, it might seem appropriate, at first sight, to modify the basic wave TLM algorithm, described above for the vibrating string, by simply adding at each time interval an extra (negative) displacement to the TLM displacement pulses corresponding to the variation in displacement due to the instantaneous value of the extra, elastic force at each point along the string. On this scheme, the displacement adjustment would be $\frac{1}{2}f\Delta t^2$, where f would be the acceleration (the elastic force, $-k\Delta xu$, divided by the mass of the element, $\rho\Delta x$), shared equally between the two pulse systems (right- and left-going).

This 'naive', direct approach, however, does not work. It fails to model the physical process correctly, and the desired modification to the TLM scheme must be developed more carefully. Why it fails should become clear as the subtleties of the correct solution schemes and their foundations are presented.

3.1. Solution Method 1: TLM pulses representing displacement, elastic force by spatial integration

Component solutions of the wave equation obey the relationship

$$u_t = -cu_x \tag{5}$$

In other words, the temporal rate of change of a wave variable is proportional to the spatial rate of change of that variable, the proportionality constant being the wave speed c. It is desired to add in a new acceleration component, that is, to change u_{tt}. This implies changing u_t over time, which in turn can be achieved by an appropriate change in u_x (see Equation (5)) at every time step. Physically, this means adjusting the spatial slope of the component waveforms so that, on propagating at the wave speed c, they produce the correct velocity change with time.

In general, however, one cannot change the gradient of a function arbitrarily throughout its range without simultaneously modifying the actual values of the entire function. In this case, modifying the entire function values would amount to instantaneously changing the shape of the string displacement throughout its length (with a view of getting the right accelerations at one point). Even if mathematically this could be repeated for every point without meeting contradictory requirements, such a procedure would make little or no sense physically.

In the present case, however, there are two counter-propagating wave components and the gradients of the *component* waves can indeed be changed in such a way that their *sum* (the actual displacement) remains unchanged. This is the basis of Solution Method 1. By suitably adjusting the gradients of the component waves, the velocity at every point can be changed continually, in proportion to the elastic force, without directly adjusting the displacement. (The subsequent displacement of the string with propagation of the wave will, of course, be affected by this process, as it ought to be, and by exactly the right amount.) It turns out that the left- and right-going waves should 'share' the required changes in gradient equally, although with opposite sign because they are moving in opposite directions.

So the first solution scheme consists of the following steps:

(a) 'Connect' (propagate) the TLM pulses for one time interval in the usual way (see below).
(b) Calculate the extra (elastic) acceleration, *hu*, at each point from the current displacement value, which in turn gives the amount by which the spatial derivative (gradient) must be changed at every point.
(c) Integrate this gradient with respect to distance along the string, adding half of this integral to the left-going TLM pulse stream and subtracting half from the right-going stream, starting at an arbitrary point (e.g. a boundary) and with a correspondingly arbitrary constant of integration (e.g. zero).
(d) Scatter and propagate the TLM pulses again in the usual way and repeat the entire cycle.

Step (c) adjusts the gradients (spatial derivatives) of the component waveforms, as required, while leaving their sum unchanged. Both the starting point for the spatial integration and the constant of integration are arbitrary, because in this TLM scheme only the sum of the component waveforms has significance. (If the same constant is added to one, and subtracted from the other, the sum remains unchanged.) In the numerical scheme, it is convenient to integrate from a boundary, with zero initial value. In the symbolic analysis below, however, it is simpler to take, as starting point for the integration in both directions, the central mesh link $i = 0$, and, zero for the initial value, so that r_0 and l_0 propagate without change at the first time step.

As a succinct way of presenting the algorithm, it is formulated below in the computer programming language 'Matlab' which is almost self-explanatory. For clarity neither initial nor boundary conditions are presented, nor is the code optimized. The 'propagate' (sometimes also called the 'connect') part of the algorithm is standard and is not shown: its effect is to shift the TLM 'right' arrays by one space increment to the right, and the 'left' arrays by one to the left. The heart of the code (after initializing variables) can be expressed as follows:

$$\text{grad\_adj} = -\tfrac{1}{2}h\,\Delta t^2 * \text{cumsum}\,(\text{u\_right} + \text{u\_left})$$

$$\text{u\_right} = \text{u\_right} - \text{grad\_adj}$$

$$\text{u\_left} = \text{u\_left} + \text{grad\_adj}$$

[Propagate displacement TLM pulses
u_right and u_left as normal, with boundary conditions]

where the variables are as follows:

- ‘u_right’ and ‘u_left’ are arrays containing the right- and left-going displacement TLM pulses, and whose sum is the displacement;
- ‘grad_adj’ is an array for (temporary) storage of gradient adjustment function;
- cumsum is a cumulative sum function, equivalent to numerical spatial integration.

3.2. Verification

The first verification is by a symbolic comparison with a finite difference approximation to Equation (1). While an implementation of the numerical algorithm is easily programmed in code, the symbolic manipulation required to verify it, in a general way, is somewhat tedious, and will be presented step by step.

The top of Table I shows a completely arbitrary initial distribution of rightward and leftward going pulses (r_i and l_i), representing displacement, at time $t_0 - \Delta t$, over five space intervals centred at link 0. The pulses are shown just after scattering at the nodes (represented by the vertical dotted lines) along a one-dimensional ‘mesh’ (solid line). The rest of the table shows some of the calculated new values over two subsequent time intervals when both propagation and slope adjustment (above steps (a)–(c), inclusive) are implemented twice. The constant a incorporates the elastic acceleration multiplier term, $-k\Delta l/\rho\Delta l$, multiplied by Δt to get the change in velocity, divided by $c = \Delta l/\Delta t$ to convert velocity adjustment to slope adjustment (cf. Equation (5)),

Table I. Method 1.

Time, $t = t_0 - \Delta t$

$r_{-2}\rightarrow$	$r_{-1}\rightarrow$	$r_0\rightarrow$	$r_1\rightarrow$	$r_2\rightarrow$
$\leftarrow l_{-2}$	$\leftarrow l_{-1}$	$\leftarrow l_0$	$\leftarrow l_1$	$\leftarrow l_2$

Time, $t = t_0$

	$r_{-2} + a(r_0 + l_0 + r_{-1} + l_{-1})$ $\rightarrow$	$r_{-1} + a(r_0 + l_0)$ $\rightarrow$	r_0 $\rightarrow$	$r_{-1} - a(r_0 + l_0)$ $\rightarrow$
$\leftarrow$	$\leftarrow$ l_0	$\leftarrow$ $l_1 + a(r_0 + l_0)$	$\leftarrow$ $l_2 + a(r_0 + l_0 + r_1 + l_1)$	

Time, $t = t_0 + \Delta t$

	$\rightarrow$	$r_{-2} + 2a\, r_{-1} + (a + 2a^2)r_0 + a\, l_{-1} + (a + 2a^2)l_0 + al_1$ $\rightarrow$	$\rightarrow$	$\rightarrow$
$\leftarrow$	$\leftarrow$	$\leftarrow$ $l_{-2} + 2a\, l_{-1} + (a + 2a^2)l_0 + a\, r_{-1} + (a + 2a^2)r_0 + ar_1$	$\leftarrow$	

multiplied by Δl for spatial integration, and finally multiplied by a half, to distribute the adjustment equally between the two component waves, giving

$$a = -\tfrac{1}{2}(k/\rho)\Delta t\,(\Delta t/\Delta l)\Delta l$$
$$= -h\Delta t^2/2$$

Whereas, in the numerical algorithms, the integration (cumulative sum) is most conveniently begun at the boundaries, here it is evaluated from the central link line, r_0, l_0, arbitrarily setting the constant of integration to zero. Thus, for example, in the first time interval shown, the right wards-going pulses, r_i, move one link length to the right and simultaneously are augmented by a times the cumulative sum from the central node (at the previous time interval), increasing leftwards, decreasing rightwards. At the first time step r_0 and l_0 therefore propagate without change, the cumulative sum being zero for them.

To avoid clutter, only a few sample terms (mainly those subsequently needed) over the next two time steps are evaluated and shown. Finite difference approximations, D_{xx} and D_{tt}, centred on the link lines at the mid-point in space and time, respectively, can then be set up for each term in Equation (1) to see how the working out of the TLM scheme compares. For example, the second time derivative, at the central TLM link-line, is approximated by a standard finite difference formulation, namely, the sum of the pulses at $(t_0 + \Delta t)$, minus twice the sum of the pulses at (t_0), plus the sum of the pulses at $(t_0 - \Delta t)$, all divided by Δt^2. This produces a long expression in terms of the original pulse values r_i and l_i, all expressed as a column in Table II. Expressions for the finite difference approximations for the other two terms in Equation (1) are similarly tabulated. It is clear that the sum of the last two columns (representing D_{xx} and $-h\Delta t^2$) equals the column representing $D_{tt}\Delta t^2$ exactly. One can therefore conclude that the TLM process matches the finite difference model of Equation (1) perfectly.

The second way of confirming the algorithm is by applying it to a problem for which an analytical solution exists. Duffy [8] uses the separation of variables approach to obtain a Fourier series solution to Klein–Gordon equation with initial displacement as shown at $t = 0$ in Figure 1 (top pair of lines) and with zero initial velocity. The solution is

$$u(x, t) = \sum_{n=1}^{\infty} \tfrac{1}{2}A_n[\sin(k_n x + \sqrt{k_n^2 c^2 + h} \times t) + \sin(k_n x - \sqrt{k_n^2 c^2 + h} \times t)] \tag{6}$$

where $k_n = n\pi/L$, L being the string length, and A_n are constants from the initial conditions.

Table II. Method 1.

	$D_{tt}\Delta t^2$	$= ?$	D_{xx}	$+$	$-h\Delta t^2$
r_{-2}	1		1		
r_{-1}	$-2 + 3a$		$-2 + a$		$2a$
r_0	$1 - 2a + 4a^2$	?	$1 - 2a$		$4a^2$
r_1	a	=	a	+	
l_{-1}	a		a		
l_0	$1 - 2a + 4a^2$		$1 - 2a$		$4a^2$
l_1	$-2 + 3a$		$-2 + a$		$2a$
l_2	1		1		

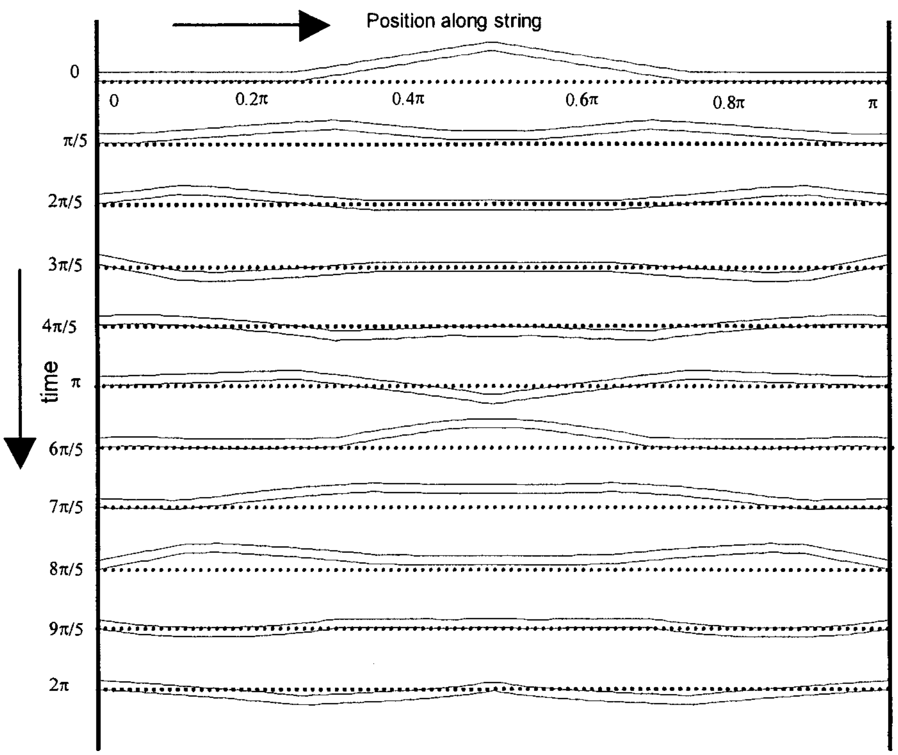


Figure 1. Fourier solution (Equation (6)) at successive time intervals and corresponding TLM solution (slightly shifted vertically upwards to allow comparison) for $L = \pi$, $c = 1$ and $h = 1$, with initial displacement as shown at the top ($t = 0$) and zero initial velocity. (See Reference [8] for full details). In fact the two solutions are indistinguishable even at moderate levels of discretization.

Figure 1 shows successive time frames of this Fourier solution paired with the corresponding TLM solution, with the same initial conditions, shifted upwards slightly to allow comparison. The horizontal axis is the string length, with reflecting boundaries (zero displacement) at each end. The initial displacement at zero velocity is shown on top and pictures at subsequent time intervals follow vertically. Any difference between the two solutions is due entirely to numerical discretization and can be made arbitrarily small by taking more terms in the Fourier series and making the TLM space and time increments finer. In the limit, the two solutions match exactly.

3.3. Solution Method 2: TLM pulses represent displacement, elastic force via temporal integration

In the second method, the effect of the elastic force on the string is modelled as if it were acting in parallel with, but almost independent of, the propagating wave motion. Over time, the elastic force causes an 'extra' acceleration (and corresponding extra velocity and extra displacement) at each point along the string, all three of which can be 'tracked' almost as if the displacement waves were not present.

In fact, this is precisely the way they *should* be 'tracked'. Equation (1) can be interpreted in this way. The acceleration term u_{tt} on the left-hand side has effectively two contributing effects shown on the right-hand side: (a) the propagating wave effect $c^2 u_{xx}$, proportional to T/ρ $(=c^2)$ and the local rate of change of the slope of the string, and (b) the non-propagating elastic effect, $-hu$, proportional to h and to the total displacement. Any interaction between these two terms on the right-hand side happens only indirectly, through the common resulting displacement. The dominance of either term can vary. One extreme case is at $h = 0$, when there is no elastic force and tension effects dominate: the displacement then propagates as with the standard wave equation. The other extreme is when $c = 0$, for example if the string tension were negligible, and there are then stationary oscillations ('elastic' or spring-like) with no propagating waves along the string. In general, the situation will be between these extremes, but still with two distinguishable contributions to the acceleration, one wave-like and propagating, the other elastic and stationary. The overall motion of the string can be described as the superimposition of these two motions, coupled only by their common displacement. In particular, the contribution to the behaviour due to the elastic force can (and should) be evaluated separately.

To achieve this independent 'tracking' in TLM, a separate array is set up to store and update the 'extra' velocity (due to the elastic force alone). The array is updated at every time interval, the local velocity change being calculated from the acceleration caused by the elastic force at each point, so that $\Delta v = [-k\Delta x u/\rho\Delta x]\Delta t$. This extra velocity gives the required, local, extra displacement $(\Delta u = v\Delta t)$ in the string. This is split equally between the right- and left-going TLM displacement pulses before being added in.

Once again, this method works well and is confirmed by comparison with finite difference schemes and by comparison with analytically derived solutions. Accuracy is found to be limited only by the level of spatial and temporal discretization.

The symbolic TLM algorithm over two time intervals (that is, at three successive times) is represented in Table III. Again a one-dimensional TLM line is represented, showing assumed, displacement TLM pulse values immediately after the scattering at nodes (vertical dotted lines) at time $t = t_0 - \Delta t$. A separate array below this (boxed with broken lines) shows the 'extra' (elastic) velocity values along the same line at the same time. To obtain the values at the next time step $(t = t_0)$, the velocities are changed (i.e. accelerated) by an amount a times the total displacement, where

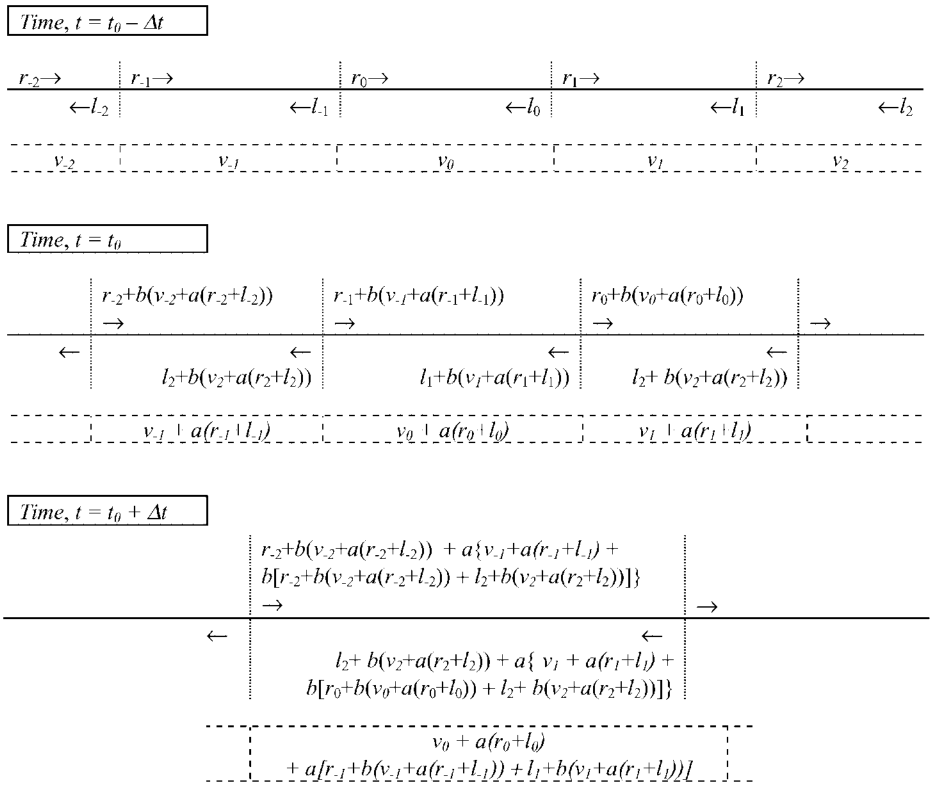
$$a = (-k/\rho)\Delta t$$
$$= -h\Delta t$$

The displacement pulses are then increased by this velocity, multiplied by the time increment to get displacement, with half of this total increment being applied to each component displacement wave. That is, the new velocity is multiplied by b, and then added to each of the counter-propagating displacement pulses, where

$$b = \Delta t/2$$

This process is repeated over two time steps. Finite difference approximations for each term in Equation (1) are then evaluated. They are expressed in terms of assumed initial TLM variable values r_i and l_i and the initial velocities v_i, and then each variable coefficient is tabulated. This is Table IV.

Table III. Method 2.



As can be seen, the *symbolic* manipulation presented in Table IV gets rather complicated (not a big problem, of course, with the actual numerical implementation). If the alignment between TLM and finite difference schemes were perfect, the sum of the coefficients in the D_{xx} and $-h\Delta t^2$ columns would equal those in the $D_{tt}\Delta t^2$ column. At first sight, therefore, the alignment is far from perfect. Bear in mind, however, that all the assumed series of initial variables (the r_i, l_i and v_i,) will be of similar magnitude within each group (that is, if wavelengths are large in comparison with mesh spacing, a standard TLM constraint). For example, r_{-2} and r_{-1} will be close in numerical value to each other. Furthermore, product terms ab ($=-h\Delta t^2/2$) will be small, for small Δt, and the squared products will be negligible. In the light of all these considerations, the finite difference and TLM algorithm solutions line up very well, with inconsistencies approaching zero as the discretization gets finer.

Slightly different tables are produced if a different order is followed in evaluating terms and updating the TLM pulses, but similar tables result. Finally, the table is symmetrical with respect to left- and right-going pulses and could be abbreviated, but it was felt simpler and clearer to leave it complete.

The ‘Matlab’ code for the heart of the algorithm is as follows:

$$\text{h\_vel} = \text{h\_vel} - h * \Delta t * (\text{u\_left} + \text{u\_right})$$

Table IV. Method 2.

	$D_{tt}\Delta t^2$	$=?$	D_{xx}	$+$	$-h\Delta t^2$
r_{-2}	$1+2ab+a^2b^2$		$1+ab$		
r_{-1}	$-2-ab$		$-2-2ab$		$2ab+2a^2b^2$
r_0	$1+ab+2a^2b^2$		$1+2ab$		
r_1	$-ab$		$-2ab$		$2a^2b^2$
r_2	$ab+a^2b^2$		ab		
l_{-2}	$ab+a^2b^2$		ab		
l_{-1}	$-ab$	?	$-2ab$		$2a^2b^2$
l_0	$1+ab+2a^2b^2$	=	$1+2ab$	+	
l_1	$-2-ab$		$-2-2ab$		$2ab+2a^2b^2$
l_2	$1+2ab+a^2b^2$		$1+ab$		
v_{-2}	$b+ab^2$		b		
v_{-1}	$-b$		$-2b$		$2ab^2$
v_0	$2ab^2$		$2b$		
v_1	$-b$		$-2b$		$2ab^2$
v_2	$b+ab^2$		b		

u_right = u_right + 1/2 * Δt * (h_vel)

u_left = u_left + 1/2 * Δt * (h_vel)

[Propagate displacement TLM pulses

u_right and u_left as normal with boundary conditions]

where

- 'u_right' and 'u_left' are arrays for the right- and left-going displacement TLM pulses;
- 'h_vel' is an array to store the 'extra' velocity due to elastic force only.

4. DISCUSSION

In Part I of this paper, two approaches for solving the Klein–Gordon Equation have been presented and each verified in two ways. Unlike the methods to be presented in Part II, both approaches use TLM pulses in the 'obvious' way, namely to represent the primary variable directly. In the case of the vibrating string, this primary variable is the local displacement from the rest position.

In both cases, integration is used within TLM to achieve the required modelling of the physical effect associated with third term in Equation (1). In Model 1, the integration is over space at each point in time. In Model 2, the integration is over time at each point in space. In both cases care is required to ensure that the integrated quantity (a) represents the effects of the 'extra' force alone, and (b) it is added to the wave variable without creating a propagating wave effect, at least not directly. In Model 1, u_x is adjusted without changing u_t directly: in Model 2, u_t is adjusted without changing u_x directly. The adjustments are 'not direct' in the sense that, once the changes have been made, they do of course have indirect effects: subsequent local displacement and wave propagation will both be different, but in the correct physical manner.

There is little to choose between the two methods. Method 1 involves more calculations at each iteration but avoids having to store an extra array for the 'elastic' velocity. Method 2 by contrast saves somewhat on calculation but requires storage of the 'extra' velocity array. In the context of present-day computer power, however, neither method has a decisive advantage.

For both methods, a symbolic verification technique has been presented which has some novel aspects.

It seems eminently feasible to extend both methods to two-dimensional problems. In two dimensions, Method 2 would probably prove simpler, as it avoids the (minor) complications associated with integrating a gradient in two dimensions.

In Part II, two further methods are presented, in which the TLM pulses no longer represent values of the primary variable in Equation (1), but are proportional to its spatial or temporal derivative, resulting in some significant advantages.

REFERENCES

1. Christopoulos C. *The Transmission-Line Modelling Method TLM*. IEEE Press, NJ & Oxford University Press: U.K., 1995.
2. O'Connor WJ, Cavanagh F. Transmission line matrix acoustic modelling on a PC. *Applied Acoustics* 1997; **50**(3):247–255.
3. O'Connor WJ. TLM modelling of acoustic-mechanical coupling in microphones. *Proceedings of the 2nd International Workshop on Transmission Line Matrix (TLM) Modelling Theory and Applications*, in collaboration with IEEE MTT Society, Technische Universität München, Munich, 29–31 October 1997; 84–90.
4. Johns PB. A simple, explicit and unconditionally stable routine for the solution of the diffusion equation. *International Journal for Numerical Methods in Engineering* 1977; **11**:1307–1328.
5. de Cogan D. *Transmission Line Matrix (TLM) Techniques for Diffusion Applications*. Gordon and Breach: Abingdon, U.K., 1998.
6. Zauderer E. *Partial Differential Equations of Applied Mathematics*. Wiley: New York, 1983; 112–113.
7. Whitham GB. *Linear and Nonlinear Waves*. Wiley: New York, 1974, *passim*.
8. Duffy DG. *Solutions of Partial Differential Equations*. TAB Books Inc.: Pennsylvania, 1968; 222–225.

AUTHOR'S BIOGRAPHY

William J. O'Connor was born in Dublin in 1951, obtained a Doctorate from University College Dublin in 1976 on magnetic fields and forces for pole geometries with saturable materials. He lectures in Dynamics, Control and Microprocessor Applications in UCD, National University of Ireland, Dublin, in the Department of Mechanical Engineering. His research interests include novel numerical modelling methods and applications, especially in acoustics, mechanical-acoustic systems and fluids; development of Transmission Line Matrix and impulse propagation numerical methods; control of elastic mechanical systems including active vibration damping; acoustic, infrared and vibration-based sensors; and analytical and numerical analysis of non-linear megnetostatic fields and forces. He is a Fellow of the Institution of Engineers of Ireland.

