The time required for simulating circuits for a given time period with TLSIM is inversely proportional to the time step used in the simulation. Larger time steps may give shorter simulation times at the expense of accuracy. This inaccuracy stems from two sources: (a) the implicit Euler method used for solving differential equations is more accurate with smaller step size and (b) the delay δ in a transmission line, in general, is not an integral multiple of the time step. The values for outgoing voltage waves at transmission line terminals are stored only for each time step. Linear interpolation of voltage wave from two consecutive samples is used in TLSIM. Fig. 5 shows the output waveform for a circuit for three different choices of step size. With decreasing step size, the improvement in accuracy becomes insignificant. Table 1 shows the variation of run times with step size for an inverter pair.

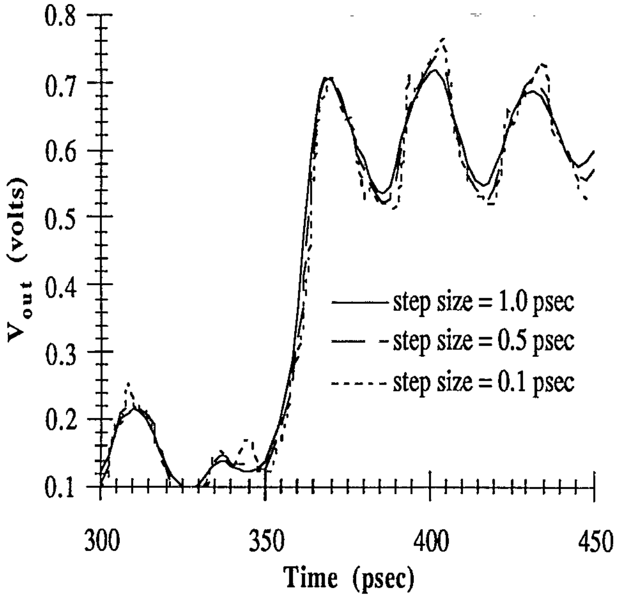

Fig. 5. Simulation accuracy vs. step size.

TABLE 1: STEP SIZE VS. RUN TIME FOR TLSIM.

STEP SIZE (psec)	RUN TIME (sec.)
1.0	0.47
0.5	0.75
0.1	2.78
0.01	20.67

Since the transmission line is modeled as delay blocks, it provides a natural way of partitioning circuits into smaller subcircuits. Due to the uncoupled nature of signals at nodes in separate subcircuits, it is possible to simulate the subcircuits separately at each time step. The computational complexity for solving the unknown node voltages of the circuit is thus reduced from $O(n^{\alpha})$ to $O(\Sigma_i m_i^{\alpha})$, where n is the number of unknown node voltages in the complete circuit and m_i is the number of unknown node voltages in the ith partition and α is the order according to which the run time increases. Depending on the numerical method used, α lies in the range 2 to 3. Short transmission lines can be introduced in large circuits without any transmission lines to reduce run time.

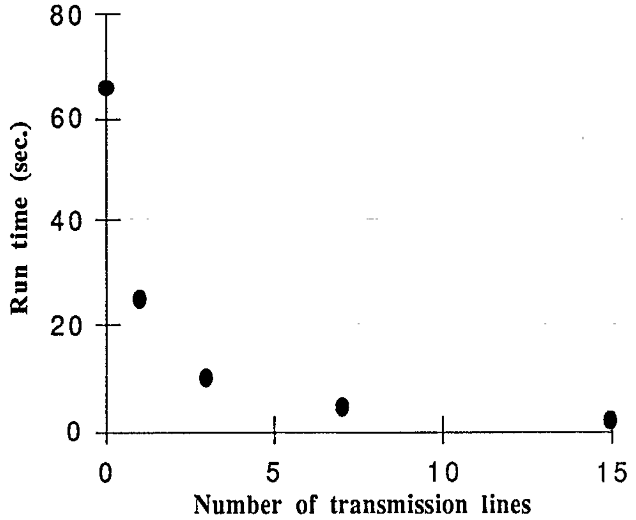

Fig. 6. Run time as a function of the number of transmission lines partitioning a circuit into subcircuits.

Run times per time step (on an IBM System 6000 model 530 machine) for simulating a 16 bit ripple carry adder with varying number of transmission lines are shown in Fig. 6. The carry adder is implemented out of GaAs DCFL nand gates. Initially, the 16 bit adder was simulated without any transmission line. Next, a transmission line is used to divide the adder into two 8 bit units. This method of partitioning is continued until 15 transmission lines are used to divide the circuit into 16 subcircuits, each subcircuit containing a single bit adder. Fig. 6 shows how run time is reduced as a result of introducing more and more transmission lines. A carry is propagated though the carry chain. The propagation of the carry starts at zero time at the primary input. The carry out at the 16th bit position is shown in Fig. 7 for two extreme cases: when no transmission line is used (solid curve) and when 15 transmission lines are used (dashed curve). The later output is a delayed version of the former output. The delay is due to the transmission lines. This delay can be seen to be negligibly small compared to the signal propagation of 1.4 ns through the carry chain.

Run time depends on the number of lines in a coupled system. Fig. 8 shows run time for simulating 16 pairs of inverters for 0.5 ns with a step size of 1 ps. Each inverter pair is connected by a transmission line. The sixteen transmission lines are divided into coupled groups, each group containing n transmission lines. Run time is shown as a function of n. Fig. 9. shows the signal on a transmission line for a window in time for n=1 (solid line), 2 and 4. The signals for n=2 and 4 are found to be almost identical.