Evolution of a non-Abelian cosmic string network

Patrick McGraw*

California Institute of Technology, Pasadena, California 91125
and Institute of Field Physics, Department of Physics and Astronomy, University of North Carolina,
Chapel Hill, North Carolina 27599
(Received 24 July 1997; published 19 February 1998)

We describe a numerical simulation of the evolution of an S_3 cosmic string network which takes fully into account the noncommutative nature of the cosmic string fluxes and the topological obstructions which hinder strings from moving past each other or intercommuting. The influence of initial conditions, string tensions, and other parameters on the network's evolution is explored. Contrary to some previous suggestions, we find no strong evidence of the ''freezing'' required for a string-dominated cosmological scenario. Instead, the results in a broad range of regimes are consistent with the familiar scaling law, i.e., a constant number of strings per horizon volume. The size of this number, however, can vary quite a bit, as can other overall features. There is a surprisingly strong dependence on the statistical properties of the initial conditions. We also observe a rich variety of interesting new structures, such as light string webs stretched between heavier strings, which are not seen in Abelian networks. [S0556-2821(98)00304-X]

I. INTRODUCTION

A generic feature of many spontaneously broken gauge theories is the existence of topological solitons, such as strings (or flux tubes), domain walls and monopoles. Many grand unified models predict the formation of such defects during a cosmological phase transition [1]. The defects could be interesting as a potentially observable signature of the symmetry-breaking pattern, and could also have important consequences for the evolution of the universe. Domain walls and monopoles, if they are stable and occur without cosmic strings, generally are not considered phenomenologically viable. Domain walls tend to produce density perturbations that are much too strong [2], whereas monopoles, if they exist at all, are predicted to form in great abundances which are incompatible with current observations [3]. Among the possible categories of stable topological defects, stringlike defects, or cosmic strings, are considered the least disastrous for cosmological models, and may be useful from a model-building point of view [4]. Among the potential applications, it has been proposed that the gravitational effects of either infinite cosmic strings or closed loops of string may serve as sources of density perturbations leading to galaxy or cluster formation [5,6].

A persistent string network could conceivably also have profound effects on the evolution of the universe due to its bulk energy density, quite apart from the effects of fluctuations. If a network of strings becomes frozen so that strings are fixed in comoving coordinates, then they will be stretched by the expansion of the universe. If the network is thought of as composed of a fixed number of segments, the number of segments per unit volume will be proportional to

a^{-3} while the length of each segment grows as a due to stretching, and so the total energy density will scale as

$$a^{-3} \times a^1 \sim a^{-2},$$

where a is the scale parameter representing the size of the universe. The energy density of nonrelativistic matter, on the other hand, scales as a^{-3}, and as the universe expands the energy in strings will grow relative to that of matter until it eventually dominates. As explained below, such a frozen network is not a typical outcome for the types of strings that have been studied to date. Instead, there are energy loss mechanisms that allow strings to be progressively destroyed, while those that survive continue to move relativistically, their total energy scaling in the same way as that of matter. It has been suggested, however, that non-Abelian strings might behave differently from Abelian ones, and might indeed lead to a string-dominated universe. A universe dominated by very heavy strings is not likely to be a viable model, but a cosmological model with a fairly recent transition to a string-dominated phase with comparatively light strings has some desirable properties [7]. Strings could serve as an interesting form of dark matter: giving density parameter $\Omega \simeq 1$ as required by inflation, while $\Omega_{matter} < 1$, and possibly resolving the apparent discrepancy between estimates of the age of the universe from the expansion rate and from stellar ages. By modifying the equation of state of the universe, they could mimic some of the effects of a cosmological constant. There has been a recent revival of interest in such a scenario [8], and testing its consistency is one of the chief motivations for the work described in this paper.

The evolution of Abelian cosmic strings has been studied extensively, and we review some of the salient points here in order to point out contrasts with the non-Abelian case. The simplest and best understood type of cosmic strings are those which occur in the Abelian Higgs model [9] and are classified by their integer winding number (an element of Z). We

*Email address: mcgraw@physics.unc.edu

shall refer to these as Z strings.[1] Assuming that only the strings with unit winding number are stable (type-II strings), they can be formed either as infinite strings or closed loops. Monte Carlo simulations [10] indicate that the infinite strings constitute a majority (63–80%) of the total length of the network initially formed by a phase transition, with the remainder comprising a ''scale-invariant'' distribution of closed loops. A very important dynamical process of these strings is intercommutation [1,11,12], the process in which two colliding strings reconnect so that a part of each string is is connected to part of the other strings. Two infinite strings intercommuting with each other twice (or a single string intercommuting with itself) can form new closed loops from pieces of the infinite strings, and this is thought to be the principal mechanism for the destruction of infinite strings.

Dimensional analysis and energy conservation arguments [4,6] provide a plausible picture of the strings' evolution: Infinite strings are destroyed by intercommutation and thus lose energy to the network of loops. Loops, in turn, shrink as they lose their energy to gravitational radiation, and they may also split into smaller loops by intercommutation. Any loop will eventually contract to a point and be destroyed. The system reaches a dynamical steady state, or self-similar evolution, characterized by a single evolving length scale which is the size of the cosmological horizon. The rate at which new strings appear within the growing horizon is balanced by the rate at which they are destroyed. An approximately constant number (order unity) of long strings stretch across a Hubble volume at a given time, and intercommutations result in the formation of a similar number of new closed loops per Hubble volume, which are destroyed within a Hubble time. While there is disagreement over details, numerical simulations [13] have supported this general picture. All results are consistent with a scaling behavior in which the energy density in strings remains a small, fixed fraction of the matter density and there is no string domination.

We emphasize that intercommutation is crucial to this picture of the strings' evolution. Since non-Abelian strings in general cannot intercommute, we might expect different behavior for such strings.

Somewhat less attention has been devoted to the evolution of branched networks, in which several strings may join at a vertex. Branched networks occur when a U(1) gauge group is spontaneously broken to Z_N with $N \geq 3$, or when the unbroken group is a non-Abelian discrete group. Among the work that has been done is that of Vachaspati and Vilenkin [14] who considered a network of Z_3 strings, which have the novel feature that three strings may intersect in a vertex. Z_3 strings tend to form an infinite network of vertices connected by string segments, with very few closed loops. It had previously been speculated that the nodes in a branched network could settle to equilibrium positions, thus causing the network to freeze as a string-dominated universe requires, but the simulations of [14] indicated otherwise. Instead of reaching an equilibrium, the nodes pull together and annihilate,

steadily reducing the number of nodes and strings. The annihilation of vertices leads to a self-similar scaling behavior, as long as the nodes are able to come close enough to each other to annihilate. The phenomenological consequences of their model are rather similar to those of Z strings: The number of vertices and string segments per horizon volume remains roughly constant, and the energy density of the network is a small constant fraction of the matter density. The strings never relax to an equilibrium, but continue to move with relativistic transverse velocities, following a self-similar evolution pattern much as that of Z strings.

The purpose of this paper is to explore the consequences of a network of non-Abelian strings. Such strings are known to exhibit a number of exotic types of interactions [15,16]. Particularly significant is the fact that when two non-Abelian strings cross each other, they cannot generally intercommute, nor can they pass through one another without forming new vertices and becoming joined by a new segment of string [17]. Linked loops of string cannot usually become unlinked, and vice versa. One might expect that this would inhibit the decay of a cosmic network by obstructing the removal of string segments. If new strings are continually being formed through string collisions, their energy must come from the already existing strings, and an equilibrium with the strings' transverse velocity damped out might seem a more likely final state. Some evidence for slowing down of the network's destruction was reported very recently by Pen and Spergel [8] in a class of models with non-Abelian global strings.

In this paper, we describe a numerical simulation of a network of S_3 strings. Unlike the authors of Ref. [8], we consider gauge strings, which have no mutual long-range interactions. Our interest is in understanding the qualitative nature of the fate of a non-Abelian string network. Do collisions result in a nondiminishing or rapidly increasing number of strings? A static equilibrium state which is expected to be conformally stretched with the universe's expansion? Does the network instead decay rapidly into finite networks and closed loops? Or does a dynamical self-similar evolution emerge, as in [14]? Another question which we hope to illuminate is: which processes play the most crucial roles in the network's evolution? The importance of intercommutation to the evolution of Z strings led to fundamental study of the dynamics of intercommutation; likewise it is hoped that the results obtained here will suggest which aspects of non-Abelian string dynamics are ripe for closer examination.

Our method of simulation is directly inspired by that of Ref. [14]: we generate initial conditions from a lattice Monte Carlo simulation and then evolve the network according to a highly simplified model of string dynamics which we hope captures the essential features of a string network losing energy. We find hints of some quite interesting physics in the interplay between the two types of string in our model, and a rather surprising dependence of the network's behavior on the initial conditions. Concerning the string-dominated cosmological scenario, we reach somewhat different conclusions from [8]. For a wide range of conditions, the network's density follows power laws very similar to the ones arising in Abelian networks. Non-abelian effects can, indeed, slow down the network's decay in the sense that they change the

---

[1]These are often referred to in the literature as U(1) strings since they result from the complete breaking of a U(1) gauge group, but it seems more logical to call them by their topological classification, as is usual for other types of strings.

coefficients of the power laws, but it appears as if the slowing down to a stable equilibrium happens only under special circumstances, if at all.

The remainder of this paper is organized as follows. Section II provides a brief summary of the properties of non-Abelian strings which bear on this simulation. We discuss some of the subtleties inherent in the description of noncommuting magnetic fluxes, and the necessity of a (gauge fixing) convention to resolve these ambiguities and allow the comparison of fluxes of strings. Most importantly, we explain why two colliding non-Abelian strings cannot, in general, cross or intercommute without forming a new segment of string. Section III describes the particular S_3 model which we have chosen to simulate. S_3, the permutation group on three elements, was chosen as the gauge group because it is a simple non-Abelian group which exhibits all of the important general characteristics of non-Abelian strings. Another motivation for this choice is that S_3 contains Z_3 as a subgroup, allowing instructive comparisons with the Z_3 network simulations of Ref. [14]. Section III also describes our procedure for simulating the network's dynamical evolution, giving enough details to allow an understanding of the results. Additional technicalities of the procedure are relegated to the Appendix. Section IV describes our procedures for generating initial conditions, and summarizes features of the networks these procedures generate. We use two different Monte Carlo algorithms which generate initial networks with somewhat different statistical properties. The network's evolution turns out to have a surprisingly strong dependence on the initial conditions. Section V presents results of the dynamical evolution simulation, exploring the influence of a number of different variables including the initial conditions of the network and the ratio of string tensions. These results are compared with those for a Z_3 network, which is Abelian. In Sec. VI, we present our conclusions and suggest directions for future work.

The Appendix describes our procedure for keeping track of string fluxes during the simulation and for establishing them from the lattice Monte Carlo procedure, covering details not included in Sec. III. The implementation of non-Abelian fluxes in a simulation presents a rather difficult problem in its own right. A careful gauge-fixing procedure is required, and some of the subtleties that arise are of interest from a field-theoretic point of view. The algorithm has been described in greater detail in Ref. [18].

In this discussion, the strings will be considered as classical objects with well-defined fluxes (after a gauge has been fixed). We will not consider quantum-mechanical effects such as Cheshire charge [19].

II. VORTICES AND STRINGS IN A NON-ABELIAN DISCRETE GAUGE THEORY

In this section, we review very briefly the definition of non-Abelian vortices and strings and some of the properties which are important for the current simulation. The formalism used here was developed for vortices in [20] and applied to strings in [16].

Generically, topological defects of codimension 1 (vortices in two space dimensions, strings in three) occur when a gauge symmetry group G is spontaneously broken to a sub-

group H such that there are noncontractible closed loops in the vacuum manifold G/H. These paths and the corresponding defects are formally classified by elements of the first homotopy group $\pi_1(G/H) \neq 1$ [1,21]. For example, in the case of a U(1) symmetry breaking completely (the Abelian-Higgs model), $\pi_1(G/H) = \pi_1[U(1)] = Z$, and strings are classified by their integer winding number. In the remainder of this discussion, we will assume, unless otherwise stated, G is simply connected and that the unbroken subgroup H is discrete. In that case $\pi_1(G/H) \sim \pi_0(H) \sim H$. No light propagating fields then remain in the broken phase, and the gauge connection is locally pure gauge everywhere outside of defect cores. If $\pi_0(H)$ is non-Abelian, then the composition of paths depends on the order. Hence the fluxes of strings will be noncommuting group elements, and that is what is meant by non-Abelian strings.

To describe the fluxes of cosmic strings, we consider closed curves Γ which lie in the (nonsimply connected) region $\mathcal{R} = \mathcal{M} - \{D\}$, where $\{D\}$ is the union of all defect cores (regions of nonvacuum) and $\mathcal{M}$ is the spatial manifold on which the defects exist. Each string gives rise to a class of noncontractible closed paths in $\mathcal{M} - \{D\}$ which encircle the string.

The flux enclosed by any closed loop Γ (e.g., one that surrounds a string) is a group element defined as a path-ordered exponential of the gauge field

$$\text{flux} = P \exp\left(\oint_\Gamma A \cdot dl \right). \tag{1}$$

For any Γ within $\mathcal{R}$ this must be an element of H. This is because the Higgs field is covariantly constant throughout $\mathcal{R}$ and so the transformation that results from parallel transport around a loop must leave the Higgs field invariant.

In a non-Abelian theory, this definition of the flux is not gauge invariant, and may depend on the point at which the path begins and ends. However, the flux through any contractible loop which does not enclose a string is necessarily trivial. A corollary of this fact is that two closed loops which share the same beginning and ending point x_0, and can be continuously deformed into each other, have the same flux. Thus the relevant structure for the description of the system of defects is the fundamental group or first homotopy group $\pi_1(\mathcal{M} - \{D\}, x_0)$, defined with respect to a base point x_0. Each string is associated with a generator of the fundamental group. Once x_0 has been (arbitrarily) chosen, the fluxes of all closed paths (and of all strings) are specified by a homomorphism from $\pi_1(\mathcal{M} - \{D\}, x_0)$ into $H \sim \pi_1(G/H)$. The only remaining gauge freedom is a global one. However, there is a considerable amount of ambiguity in what we mean by ''the flux'' of one particular string: an arbitrariness in how exactly the set of generators is chosen for the homotopy group $\pi_1(\mathcal{M} - \{D\}, x_0)$. In Fig. 1, for example, there are two loops, both beginning and ending at x_0, both enclosing the same string without enclosing any others, which are nonetheless representatives of different homotopy classes (and consequently may be associated with different fluxes). An intervening string prevents one path from being continuously deformed to the other. The fluxes associated with the two different paths may differ through conjugation by the flux of the other string. In an Abelian theory, conjugation is trivial;
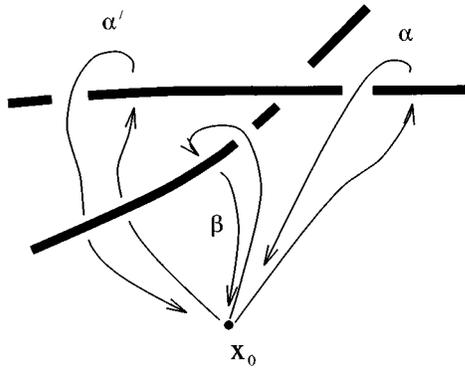
FIG. 1. The paths α and α' both enclose the same string and no other strings, but they cannot be continuously deformed into each other without crossing another string. Thus, they represent different elements of the fundamental group $\pi_1(\mathcal{M}-\{D\},x_0)$, and so the fluxes associated with them may be different. Specifically, the homotopy classes of α and α' are related through conjugation by another generator: $\alpha' \sim \beta\alpha\beta^{-1}$. (We follow the usual convention of composing paths from right to left: $\beta\alpha\beta^{-1}$ means the path formed by traversing first the reverse of β, then α, then β. The relation $\sim$ represents homotopy equivalence.) The associated fluxes are analogously related: a nontrivial relation if the fluxes do not commute.

not so in a non-Abelian one. It follows that fluxes cannot meaningfully be compared (say, to determine if they are the same) if the paths used to define those fluxes pass on opposite sides of some other string. Comparisons must be made using ''nearby'' paths.

Since the flux of the same string may described, depending on convention, by distinct conjugate elements of H, it follows that strings whose fluxes are in the same conjugacy class must be generate in tension. It is not true, however, that all fluxes in the same conjugacy class are the same. If a and b belong to the same conjugacy class, it is by no means guaranteed that a^2 and ab also are conjugate, or that $ab = ba$. Distinctions among elements within the same conjugacy class can have important consequences in any situation where the product or commutator of two fluxes is relevant, as is the case when two strings collide.

The path dependence of the flux of a string implies an important fact: two strings with noncommuting fluxes cannot pass through each other without forming a new segment of string whose flux is the commutator of the fluxes of the two original strings. Penetration without the formation of a new string would violate flux conservation [17]. This is illustrated in Fig. 2. Noncommuting strings also cannot intercommute. We will be especially interested in the consequences of this entanglement process for the evolution of a string network: it might impede the collapse of the network.

III. OUR MODEL AND ITS DYNAMICS

S_3 *strings.* We consider here a model with unbroken gauge group $H = S_3$, the permutation group on three objects. The spectrum of this model will include strings whose fluxes are elements of S_3. S_3 has six elements in all. The identity e corresponds to the trivial permutation. There are three odd permutations (two-cycles or transpositions) each leaving one
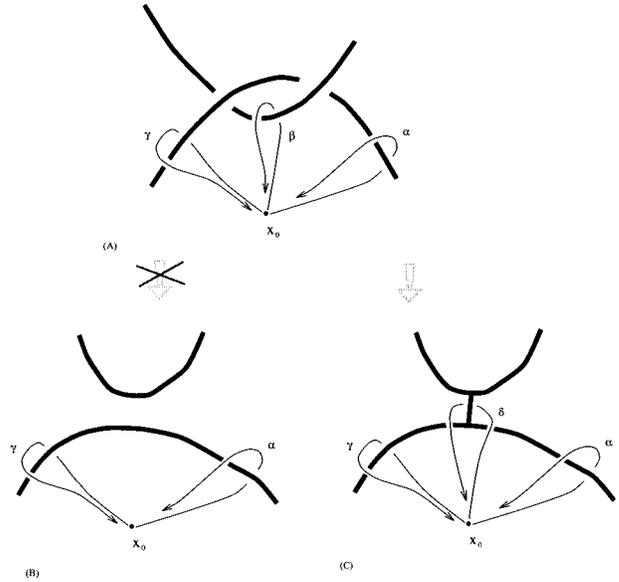

FIG. 2. Attempt to pass two strings through each other. In (A) the flux of one string may be defined by either of the paths α or γ, and that of the other string by β. Let the fluxes associated with α, β, and γ be a, b, and c, respectively. In this case, $c = bab^{-1}$. In general, $c \neq a$. Now, if we attempt to pass the strings through each other, no strings need cross the paths α and γ, so the associated fluxes will not change. But if the strings were to pass through each other freely, as in (B), α and γ would be continuously deformable into each other. This is impossible if they have different fluxes. In order to conserve flux, the string must branch somewhere and be connected to the other by a new string whose flux as defined by path δ in (C) is $ca^{-1} = bab^{-1}a^{-1}$.

of the three elements invariant and interchanging the other two. We may denote these, for convenience, by $t_1 \equiv \{(123) \rightarrow (132)\}$, $t_2 \equiv \{(123) \rightarrow (321)\}$, $t_3 \equiv \{(123) \rightarrow (213)\}$. In this notation, t_i is the two-cycle which leaves the ith element in the same position. The two nontrivial even permutations are the three-cycles, or cyclic permutations, which we denote here by $s_+ \equiv \{(123) \rightarrow (312)\}$, $s_- \equiv \{(123) \rightarrow (231)\}$. In the more conventional cycle notation [22], we have $t_1 = (23)$, $t_2 = (13)$, $t_3 = (12)$, $s_+ = (123)$, $s_- = (132)$.

The two-cycles form one of the two nontrivial conjugacy classes, and the three-cycles form another. Thus our model supports two types of strings, which we shall refer to as odd and even strings, or alternately as t and s strings. The three-cycles generate a Z_3 subgroup, so that our model contains the Z_3 model as a subset. Three even strings may meet at a vertex, just as in the Z_3 model. Another type of junction is one where two two-cycle (or odd) strings merge to form a three-cycle (even) string. Figure 3 shows the two types of junctions in our model.

Since each two-cycle is equal to its inverse, oppositely oriented odd strings are topologically equivalent. An even string, on the other hand, possesses a natural orientation: The flux through a path encircling it with one orientation is s_+, while it is s_- for the opposite orientation. In subsequent figures, even strings will often be denoted by oriented lines, with the string carrying flux s_+ in the direction of the arrow,
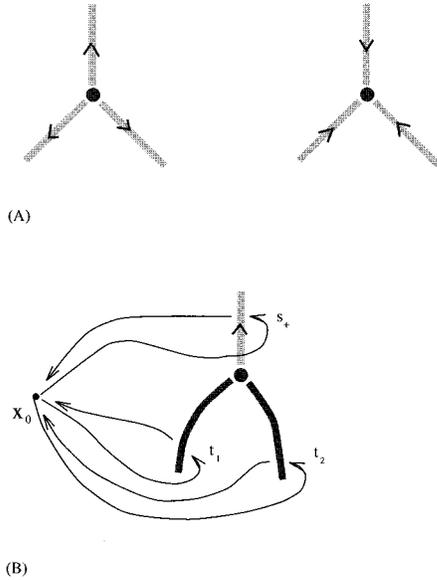
(A)

(B)

FIG. 3. String junctions in the S_3 model. (A) Two possible sss junctions: three strings with the same flux, s_+ or s_-, emanate from the node. (Or two s_+ strings merge into a single s_-, etc.) (B) One of the class of stt junctions: Two t-strings merge into an s string. Fluxes are defined with respect to a base point x_0 by the paths shown. Here, as in many subsequent figures, an s string is drawn as an oriented line. The string carries flux s_+ in the direction of the arrow; i.e., a counterclockwise path around the arrow encloses flux s_+.

while odd strings have no arrow, reflecting their lack of orientation.

Note that the parity is conserved at any junction: i.e., an odd string entering a junction must be matched by another odd string leaving. In this sense, odd strings can never end, even if they change their flux at a junction. Even strings, on the other hand, may end at a junction. It is helpful to view the network as being composed of two interacting subsystems. One subsystem consists of infinite or closed t strings (as with Z strings, they have no free ends). The other comprises a branched Abelian web of s strings, some of which end on t strings. The simulation results presented later in this paper exhibit an interesting interplay between these two subsystems.

String dynamics. The system we simulate consists of three-way junctions, or nodes, joined by s and t strings. The strings are approximated as straight segments between junctions. In effect, we are averaging over transverse oscillations of the strings. However, it is possible for a string segment to be interrupted by a pair of nodes doubly linked to each other as in Fig. 4. Doubly linked nodes tend to annihilate fairly quickly, and such configurations play the role of transient kinks on otherwise straight segments.

As in [14], the nodes are assumed to undergo damped motion under the influence of string tensions. The energy loss leads to the shortening of strings. As an approximation for the energy loss of a real string network, the model of damped motion of the nodes is most realistic if one supposes that the string junctions are monopoles which carry some unconfined magnetic flux, as in a model with the symmetry-breaking pattern

FIG. 4. Doubly linked nodes. A pair like this can interrupt an otherwise straight string or act as a kink.

$$G \rightarrow \frac{S_3 \times [SU(3) \times SU(2) \times U(1)_{EM}]}{D},$$

where D is a discrete factor divided out so that the monopoles at string junctions may carry electromagnetic U(1) charge. Their magnetic charges should then result in radiation damping.[2]

Our simulation proceeds in discrete time steps. During each time step, each node is moved by a displacement proportional to the vector sum of all tensions acting on it. This type of evolution corresponds to damped motion force $\propto$velocity. The constant of proportionality is a parameter which may be absorbed into the size of the time step. Thus, in appropriate units,

$$\Delta \mathbf{x} = \Delta t \sum_r T_r \mathbf{n}_r, \tag{2}$$

where Δt is the time step, $\mathbf{n}_r$ is the unit vector along the direction of the rth string connected to the node, and T_r is the magnitude of that string's tension. r runs from 1 to 3 for the three strings that meet at each node. Since our model has two types of string with possibly different tensions, the ratio of these two tensions is a variable parameter of the simulation. When we present the results in Sec. VI, we use units such that the lattice spacing of the initial-condition Monte Carlo algorithm is 1 and the tensions are of order unity. (More specifically, the t-string tension is normalized to 1 while the other tension is varied.)

Each node is moved in turn. During the motion of a node, the strings attached to it may collide with other strings. The annihilation of two nodes is allowed if they approach each other more closely than a certain distance d_{ann}. The procedures for handling collisions of strings or nodes are as follows.[3]

(i) Intercommutation. If, in the process of moving a node from its initial to final position, one of its string segments
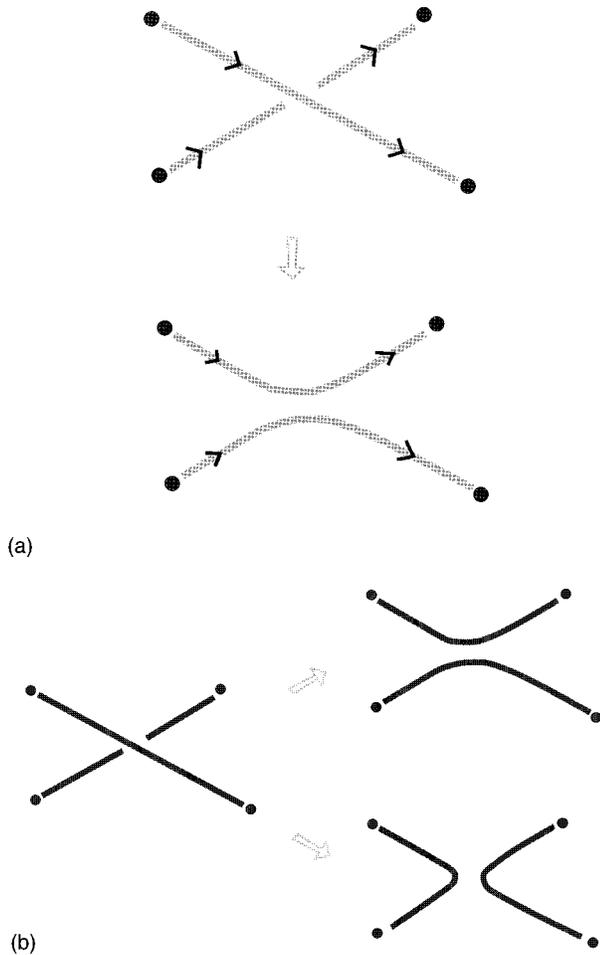

(a)

(b)

FIG. 5. (a) When two three-cycle or s strings intercommute, there is a unique rearrangement which is compatible with the orientations of the strings. (b) Since t strings (two-cycle strings) have no orientation, an intercommutation can result in either of two possible rejoinings of the ends.

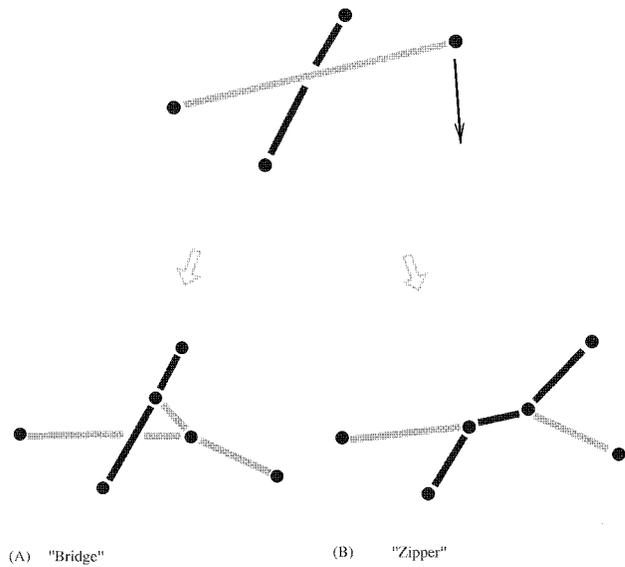(A) "Bridge"   (B) "Zipper"

FIG. 6. The intersection of two strings whose fluxes do not commute causes them to become linked by a new segment in the "bridge" configuration (A). Alternatively, they may coalesce along part of their length, forming a "zipper" (B).

intersects some other segment, then the fluxes of those two segments are compared at the point where the crossing occurs. If the fluxes commute, then the two segments may either pass through each other unaltered, or intercommute. The probabilities of these two outcomes may in principle be taken as an adjustable parameter of the simulation, but we have chosen to let the probability of intercommutation be 1 in all cases. It is widely believed that intercommutation is generically the more common outcome whenever two cosmic strings cross. In the self-dual limit, it can be shown that Nielsen-Oleson strings always intercommute. Therefore the choice to set the intercommutation probability to 1 seems a natural one.

Intercommutation may occur in two possible situations: either both strings are three-cycle (s) strings, or both are two-cycles. In the latter case, the fluxes of the two strings must in fact be equal. In an intercommutation, the string ends are rearranged in such a way as to conserve flux. In the case of two s strings, there is always only one way to rearrange the ends, as shown in Fig. 5(a). A string end carrying flux s_+ to the point of intersection may not be joined to one carrying the inverse flux s_-. When two t strings intercommute, how-

ever, there are two possible rearrangements of the ends, owing to the fact that a two-cycle is equal to its own inverse and two-cycle or t strings consequently have no preferred orientation[4] [Fig. 5(b)]. In the absence of a reason to prefer one of these rearrangements over the other, the choice is made randomly.

When an intercommutation occurs, we join the segments with a pair of "kinks" (doubly linked nodes as in Fig. 4) which will later annihilate and allow the segments to straighten. By delaying the straightening of the rejoined segments, we prevent the highly noncausal cascades of intercommutations which might otherwise occasionally occur within a single time step and lead to computational infinite loops and other unpleasant consequences.

(ii) Noncommuting collisions (NCC). If two noncommuting strings intersect, then it is assumed that they form a new pair of nodes and thus become linked by a new segment. This may happen in two possible ways. The two strings may pass through each other and become linked by a new segment which stretches between them, as shown in Fig. 6(a). We refer to this outcome as the "bridge" configuration. The flux of the intervening string segment is uniquely determined by the requirement of flux conservation. (The intervening flux of the bridge must always be a three-cycle, as the commutator subgroup of S_3 is Z_3.) Another possible outcome, consistent with the topology, is that the two colliding seg-

---

[4] Strictly speaking, we can only say that there is no *topological* reason for a t string to have a preferred orientation. It is possible that the field equations could have two distinct solutions, corresponding to differently oriented strings, which are topologically equivalent but can be deformed into one another only by surmounting a finite energy barrier. A situation of this sort occurs in the global vortices of nematic liquid crystals. This was pointed out to me by Preskill.
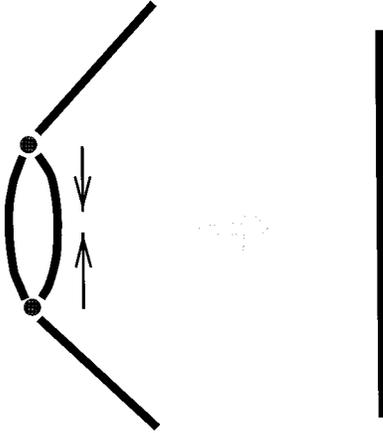
FIG. 7. A pair of nodes linked by two strings may annihilate, leaving a single string.

FIG. 8. Annihilation of two sss nodes joined by a single string. There are two possible ways to reconnect the strings consistent with their orientation.

ments stick together in the ''zipper'' configuration shown in Fig. 6(b). The possibility of collisions resulting in zipper formation have recently been mentioned in the context of type-I Abelian strings [26]. Which of these two outcomes is more likely is a dynamical and kinematic issue to be addressed in future work. It may depend on the relative velocities and angles of the colliding strings. Short-range forces between strings may also affect the growth of zippers, especially for low-angle, low-velocity collisions. Attractive interactions make zipper growth more likely and more rapid than otherwise expected. In the present simulation, we consider both possibilities separately and examine their consequences: some runs were performed with only ''bridge'' NCC's, and some with only ''zipper'' formation. In fact, we find that the choice makes little difference for the network's evolution.

Whenever two new nodes are created by a NCC, we separate them immediately by a small initial distance comparable to the time step and subsequently allow them to move normally under the influence of string tensions. String tensions may cause a zipper or bridge to grow longer after it is formed.

(iii) Annihilation. When two nodes approach each other within a distance d_{ann} which is a parameter of the simulation, they are allowed to annihilate. (In this work, $d_{ann}=0.08$ was chosen.) The segment(s) which join the two nodes is eliminated, and the other segments emanating from the two annihilating nodes are joined to each other.

Two nodes are able to annihilate only if there is a consistent way to rearrange the free string ends (i.e., each string is able to find a partner with the same flux). Annihilation is always possible if the two nodes are doubly linked as shown in Fig. 7. In the case of double link annihilation, the segment is straightened (or straightened until an obstruction is encountered, such as a collision with another string.) Annihilation is also always possible if both junctions are of the sss type, even if they are only singly linked. In this case, there are two possible rearrangements of the free string ends (Fig. 8). One of these two is chosen at random. When two stt-type junctions approach each other, on the other hand, there can be at most one consistent rearrangement of the free ends, and it may not be possible for the nodes to annihilate at all. Annihilation re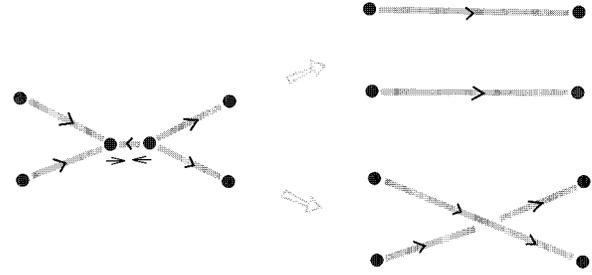quires that *each* of the two segments on one side be matched with one on the other side carrying the *same flux*. Figure 9 shows an example of a pair of nodes which cannot annihilate because there is no consistent rejoining of the string ends.

If two stt-type nodes do annihilate, it is easily seen that there can never be more than one consistent rearrangement of string ends. If two of the outgoing ends are s strings and two are t strings, then there cannot be more than one rearrangement because each string can only be joined with one in the same conjugacy class. If all outgoing strings are of odd type, then all four cannot have the same flux—if they did, then the total flux of any pair would be trivial and they would not be connected by a segment. Nor may any three have the same flux. It follows that, at best, each string end may reconnect with a unique partner. When a singly linked pair of nodes annihilates, the segments are first rejoined in a kink configuration, which may straighten later.

Another type of annihilation process, which is the inverse process of bridge formation, is not included explicitly in the simulation but may occur through a multistep process involving several string intersection and annihilation events (Fig. 10). We expect that such a process probably *will* occur whenever the geometry is appropriate for the unlinking of two strings, so that it is not necessary to perform the unlinking ''by hand'' in a single step within the simulation.

(iv) Rearrangement. If two nodes approach each other closely but are prevented from annihilation by flux conservation requirements, several outcomes are conceivable. They may stick together and form a stable junction of more than three strings. They may bounce back and move apart again under the influence of string tensions. It is also plausible that the nodes could rearrange their connections and undergo a sort of quasi-intercommutation, as shown in Fig. 11. In this simulation, we allow the nodes to bounce by introducing a small repulsion at distances shorter than d_{ann}. We also allow rearrangement with some probability, and examine the consequences of setting that probability either to zero or to some
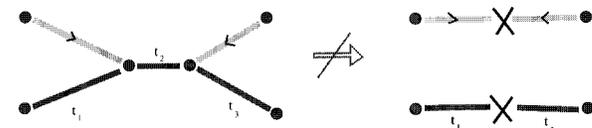

FIG. 9. The two nodes shown here cannot annihilate, because there is no consistent way to reconnect the string ends.
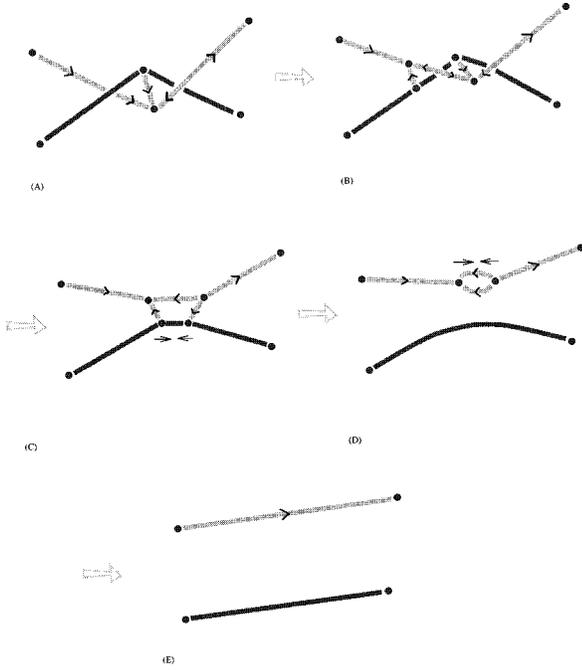
FIG. 10. Unlinking of two strings—the inverse of the process shown in Fig. 6—can occur in several steps if the string tensions pull in the right direction to unlink the strings. A linking followed by two annihilations has the net result of removing the short intervening segment and unlinking the two longer strings. In this figure, the base point is assumed to lie behind the page, so that the definition of an s string's flux changes when it passes in front of any t string.

nonzero value. The results presented in Sec. VI indicate that rearrangement speeds up the network's decay.

We do not include nodes of more than three strings as fundamental objects, but it is quite possible for a pair to become stuck together very close to each other. The string tension and short distance repulsion allow them to oscillate at a short distance, and such a configuration can behave like a single junction of four strings. Such adhesion becomes important under certain conditions, as we will see in Sec. VI.

IV. INITIAL CONDITIONS

In order to perform our dynamical simulation, we must start with an initial configuration. The generation of initial conditions models the symmetry-breaking transition which produces the strings. As is frequently done, we use a lattice
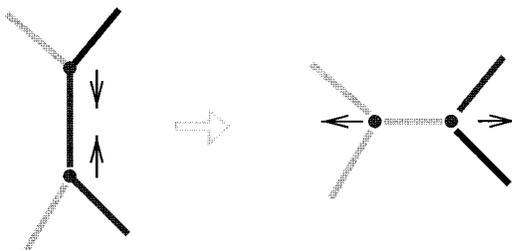

FIG. 11. Rearranging the connections of two nodes which cannot annihilate.

Monte Carlo procedure to generate an initial string network. The lattice spacing is to be identified with the correlation length of the Higgs field at the time of string formation. The Higgs vacuum expectation value (VEV) is thus uncorrelated over distances longer than a lattice spacing, and at each site of the lattice, it takes a random value within the vacuum manifold. With a suitable interpolation along the length of each link, any plaquette of the lattice is mapped to some closed loop on the vacuum manifold. If this path is one of the noncontractible loops, then a string must pierce the plaquette. Each link of the plaquette is associated with an element of G which relates the Higgs values at the two ends of the link. The product of these elements around a closed plaquette must lie within the unbroken group H, and can be taken as the flux of the string which pierces that plaquette. Strings which pierce the faces of a given unit cube must be joined together inside the cube in some way. If only two faces of the cube have nontrivial flux, then we interpret this as a single string segment passing through the cube. If three faces are nontrivial, we infer that there is a single vertex inside the cube. Cubes pierced by more than three ends require a more complicated arrangement of nodes and strings inside the cube. There may be more than one consistent way to join the string ends, and one must be chosen arbitrarily.

Two different lattice Monte Carlo algorithms have been used for the current simulation to generate different initial distributions of strings. The first, very simple way to generate a random network of strings is to use an infinite temperature lattice gauge theory: simply assign a random element of the unbroken group $H = S_3$ to each link of the lattice, and evaluate the product of links on the plaquette to find the flux through the plaquette. We refer to this as the lattice gauge Monte Carlo. There is no direct reference to a Higgs field in this technique.

The other method we use is a discrete Higgs simulation analogous to that of Refs. [27,28]. The essence of this method is that a discretized vacuum manifold is used. The breaking of continuous group G is modeled by using some discrete subgroup $\mathcal{G} \subset G$ which contains the unbroken group H. Each lattice site is then assigned randomly to one of the discrete cosets in the space $\mathcal{G}/H$, corresponding to a choice of vacuum. With each link of the lattice there is associated an element of $\mathcal{G}$ which transforms the Higgs field value at one end of the link to the value at the other. The element relating one coset to another is not unique; the possible elements themselves form a coset. The convention in this discrete Higgs method is to choose the "smallest" possible element for each link variable. "Small" is defined with reference to a metric on the continuous group G: if all elements are written in the form $g = \exp(i\alpha T)$, where T is a normalized element of the Lie algebra of G, then the smallest element is the one with the smallest number α. In this way the Higgs field is effectively interpolated in the smoothest possible way between lattice points.

A suitable gauge transformation can be performed so that all Higgs field values lie in the same coset, and all link variables lie within H, allowing all subsequent computations to be performed in terms of only H link variables. This is a lattice implementation of unitary gauge.

For the present simulation, we take $\mathcal{G}$ to be one of the discrete subgroups of SU(3). The simplest choice is the 24-

element ''dihedral-like'' subgroup of SU(3) known as $\Delta(24)$ [29]. This group is generated by the matrices

$$T_1 \equiv \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}, \quad S_+ \equiv \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix},$$

and

$$A(1,0) \equiv \mathrm{diag}(1,-1,-1).$$

$\Delta(24)$ is the smallest subgroup of SU(3) that contains S_3, and in fact it is isomorphic to S_4, the permutation group on 4 elements: an isomorphism may be defined which maps

$$T_1 \leftrightarrow t_1 = (23), \quad S_+ \leftrightarrow s_+ = (123);$$

these generate the subgroup $H_0 \simeq S_3$ of permutations on elements 1–3. H_0 may be viewed as the little group of a ''Higgs'' VEV which has the form $h_0 \equiv (0,0,0,1)$: permutations of the first three positions leave h_0 invariant. The elements $A(1,0) = \mathrm{diag}(1,-1,-1)$, $A(0,1) = \mathrm{diag}(-1,-1,1)$, and $A(1,1) = \mathrm{diag}(-1,1,-1)$ may be mapped to S_4 by

$$A(1,0) \leftrightarrow (14)(23), \quad A(0,1) \leftrightarrow (24)(13),$$

$$A(1,1) \leftrightarrow (34)(12).$$

These act nontrivially on h_0, and together with the identity they generate the four distinct left cosets H_0, $A(1,0)H_0$, $A(0,1)H_0$, and $A(1,1)H_0$. Each of the three nontrivial cosets consists of the set of elements which transform h_0 to one of three other possible VEV's. For example, elements of $A(1,0)H_0$ take $(0,0,0,1)$ to $(1,0,0,0)$.

To generate a network of strings, the discretized Higgs VEV is randomly assigned to one of its four values at each lattice site. For a smooth interpolation, the link between two neighboring sites is chosen to be the smallest within the appropriate coset. In the coset $A(1,0)H_0$, for example, the two smallest elements are $A(1,0)T_2(14)(23)(13)$ and $A(1,0)T_3(14)(23)(12)$. Both of these have equal measure; a random choice may be made between them. In the identity coset, of course, the identity element is the smallest. Each other coset has two smallest elements of the form AT, where $A \in \{A(1,0),A(0,1),A(1,1)\}$ and T is one of the two transpositions in S_3 that fail to commute with A.

After assigning vacuua and group elements, one can then transform to the unitary gauge in which the Higgs VEV is the same at each site, and all flux information is encoded in S_3 variables on the links, just as it is in the lattice gauge Monte Carlo.

Properties of the initial network. Both of the Monte Carlo algorithms described above create infinite branched networks. The lengths of string segments between branching are distributed exponentially, reflecting a constant probability of branching per unit length. In this respect, the two methods are similar, but the resulting networks differ in other statistical properties. In Table I, we summarize some of these features. For comparison, we also include the corresponding information for the Z_3 system (including both a Z_3 lattice-gauge method and the tetrahedral discrete-Higgs simulation of [28]). The Z_3 discrete-Higgs numbers are from Ref. [28]
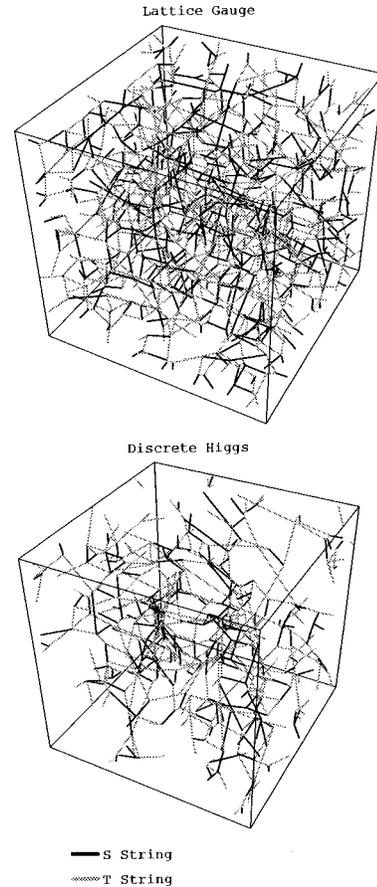

FIG. 12. Sample Monte Carlo initial configurations (actually shown after a single time step of dynamical evolution). The even, or three-cycle, strings are shown in the darker color. Volume shown is 8^3 in lattice units.

and the Z_3 lattice gauge data are based on the author's simulations (see also [30]). For each method, the fraction of plaquettes pierced by strings of each type is reported. (In the Z_3 case, of course, there is only one type.) Note that in the lattice gauge method, each group element is weighted equally; therefore $\frac{2}{3}$ of all plaquettes are pierced by strings in the Z_3 lattice gauge case, and $\frac{5}{6}$ for S_3. Below this are the fractions of cubic lattice cells with 0, 2, 3, 4, 5, and 6 of their faces pierced by strings, the number densities of the different types of nodes per unit volume, the average length between junctions and the branching length (obtained from the exponential decay of the length distribution) for each string species.[5]

Pictures of typical initial string networks (Fig. 12) illustrate qualitatively the comparison between the different S_3 initial conditions. It is evident that for both Z_3 and S_3 systems, the lattice gauge method produces a denser network: more cube faces are pierced by strings and more cells have

_____

[5]Initially, string lengths are naturally clustered near integer multiples of the lattice spacing. The distribution looks smooth and exponential only when string lengths are placed in bins of at least one lattice spacing. For this reason, the decay length is not necessarily identical to the average segment length.

TABLE I. Statistics of initial conditions.

Method	Z_3 L.G.	Z_3 Higgs	S_3 L.G.	S_3 Higgs
Faces with				
s string	0.67	0.52	0.33	0.14
t string			0.50	0.37
Cubes with				
0 ends	0.01	0.04	0.00	0.06
2 ends	0.12	0.34	0.01	0.32
3 ends	0.17	0.20	0.05	0.21
4 ends	0.38	0.32	0.21	0.31
5 ends	0.24	0.09	0.20	0.02
6 ends	0.09	0.02	0.55	0.08
Density of				
sss nodes	0.56	0.28	0.22	0.03
stt nodes			1.36	0.67
Av. length bet. junctions				
s string	2.38	3.71	1.02	1.27
t string			1.05	1.59
Branching length				
s string	1.65	3.33	0.69	0.83
t string			0.78	1.53

FIG. 13. Number of nodes and total string length as a function of time for Z_3 strings on a 30^3 volume.

high numbers of strings emerging through their faces. Correspondingly, in the lattice gauge method, fewer strings continue through more than one lattice cell without branching. Regardless of the initial conditions, stt junctions outnumber sss junctions in the S_3 network, and we can infer that a majority of the s strings end on a t string at at least one end.

Although we have not extracted detailed statistics on the presence of disconnected loops and finite networks in the initial distributions, they do not appear to form a significant part of the system. In this respect the S_3 networks are similar to the Z_3 ones. The results presented in the next section show that in the S_3 model, unlike the Z_3, differences in the statistical properties of the initial networks can have a very pronounced effect on the network's evolution.

V. DYNAMICAL RESULTS

In this section, we present results from the dynamical simulation. The aim is to give a qualitative picture of the evolution and to determine which factors are most important in determining the fate of the network. The qualitative nature of the evolution depends on the ratio of the different string tensions. Two contrasting regimes are of interest: one in which the even strings are light, and another in which they are heavy. We also examine the influences of other factors. The results show a surprisingly marked dependence on statistical properties of the initial network.

This section is organized as follows: First, we review some features of the behavior of Abelian, Z_3 strings, for comparison with the S_3 results. The Z_3 results are taken in part from [14] and in part from simulations by the author. The self-similar scaling evolution and the associated power-law behavior are demonstrated. After this come the S_3 re-
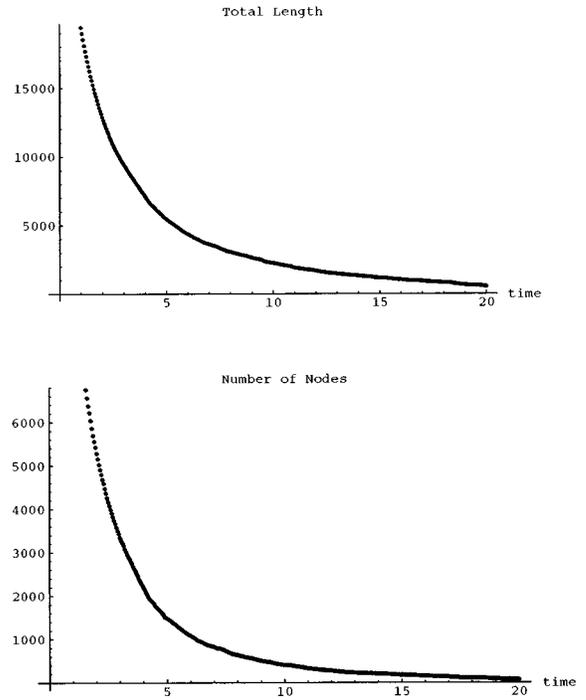
sults. We describe the dependence of these results on parameters of the simulation, focusing especially on two contrasting regimes of string tensions, and then discussing other important factors. Finally, we make a few comments about fluctuations and the effect of the finite simulation volume.

The Z_3 network and power-law evolution. Since S_3 contains Z_3 as a subgroup, our program can easily be modified to simulate a Z_3 network by generating only s strings in the initial conditions. Figure 13 shows the total string length and total number of nodes as a function of time for a typical run on a 30^3 simulation volume. The initial conditions were generated with a Z_3 lattice gauge method, rather than the discrete Higgs method of [14]. (In fact, these data were obtained in a simulation without any intercommutations—strings were allowed to pass through each other. There is no topological obstruction to prevent this in an Abelian network, and the inclusion of intercommutations makes only a small difference in the results.) The units used for this plot are the ones in which the initial lattice spacing is 1 and the string tension is $T=1$.

The time variable plotted on the x axis in the figure is $i\Delta t$, where i is the number of elapsed time steps. (This will be our convention for all remaining plots.) All distances and lengths are measured in units of the original lattice spacing.

A transformation of the data shows more clearly the "scaling" behavior of the network. The scaling hypothesis is that the gross properties of the network are described by a single length scale ℓ, which grows with time as the network relaxes. If ℓ is the typical distance between nodes, then the number of nodes per unit volume is $n \sim \ell^{-3}$. In a scaling solution, the average length of a string segment between nodes is also $\sim \ell$, while the number of segments per unit volume is proportional to the node density ℓ^{-3}. Hence the
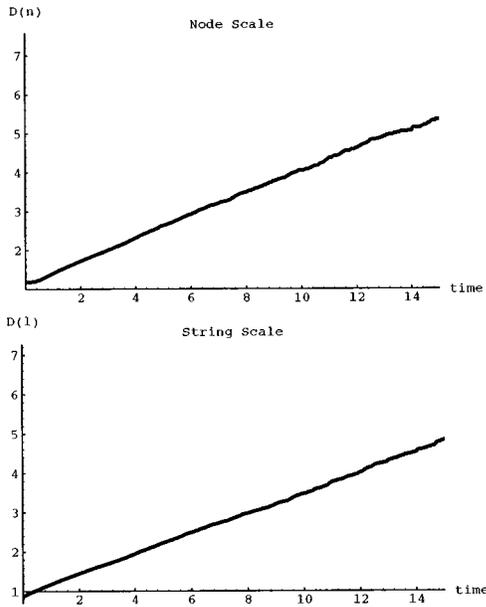
FIG. 14. Scaling behavior of Z_3 strings on a 30^3 volume. The typical distance between nodes D_n grows linearly with time, and so does the inverse square root of the string energy density D_l.

total string length per unit volume (or the energy stored in strings) scales as ℓ^{-2}. In Fig. 14, we plot two scale variables with the dimensions of length: The inverse cube root of the number of nodes per unit volume, which we denote D_n, and the inverse square root of the string length per unit volume D_l. From Fig. 14, we can see that both of these length scales are approximately equal and grow linearly with time. The slope of approximately 0.3 is close to that observed in Ref. [14]. Although the results plotted in Fig. 14 and those of [14] were obtained from different initial conditions, the results agree very closely. Evidently, all noticeable differences disappear after just a few time steps. As with Z strings, the late-time evolution is essentially independent of the initial conditions.

Evolution of the Z_3 network is self-similar in the sense that network looks statistically the same at all times except for the increase in scale. The distribution of string segment lengths, for example, is exponential at all times, with only the scale changing. This can be seen in the semilogarithmic histograms of Fig. 25. Qualitatively, a portrait of part of the network at a given time is indistinguishable from a suitably magnified portrait of an earlier time.

S_3 *network: general comments.* In the remainder of this section we describe results of the S_3 simulation, which was run using a variety of different combinations of initial conditions, string tensions, and other parameter choices. Networks generated by lattice-gauge initial conditions were run on an 8^3 simulation volume, while the less dense discrete-Higgs-generated networks were evolved on a volume of 12^3. Statistical variables were obtained from averages over several runs. Computation time constraints made it unfeasible to run many times on larger volumes, but a few runs were performed on both larger and smaller volumes in order to examine the effects of finite size.

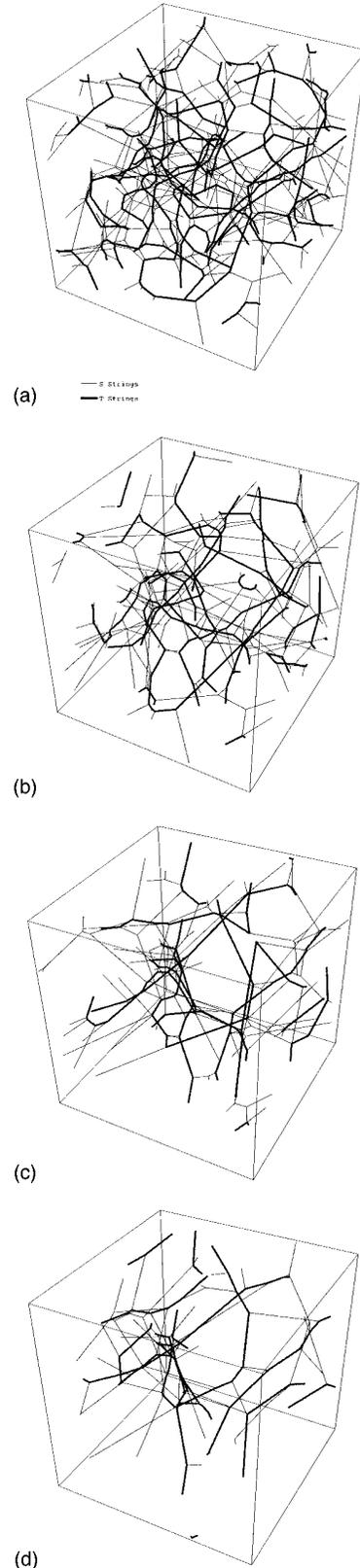Key results are displayed graphically in Figs. 15–25 and Table II. Some results for an Abelian, Z_3 network are also

FIG. 15. (a) A series of snapshots showing the evolution of an S_3 network with light strings. The heavier, odd strings are shown in darker color. We can observe long strings straightening and loops of t string contracting. The initial network was generated by the lattice gauge Monte Carlo algorithm: it is the one shown in Fig. 12. The full 8^3 simulation volume is shown. In the first frame, at $t=2$, the odd strings are quite crumpled and are connected by a dense web of even strings. (b) $t=4$. (c) $t=6$. (d) $t=8$.

(a)

(b)
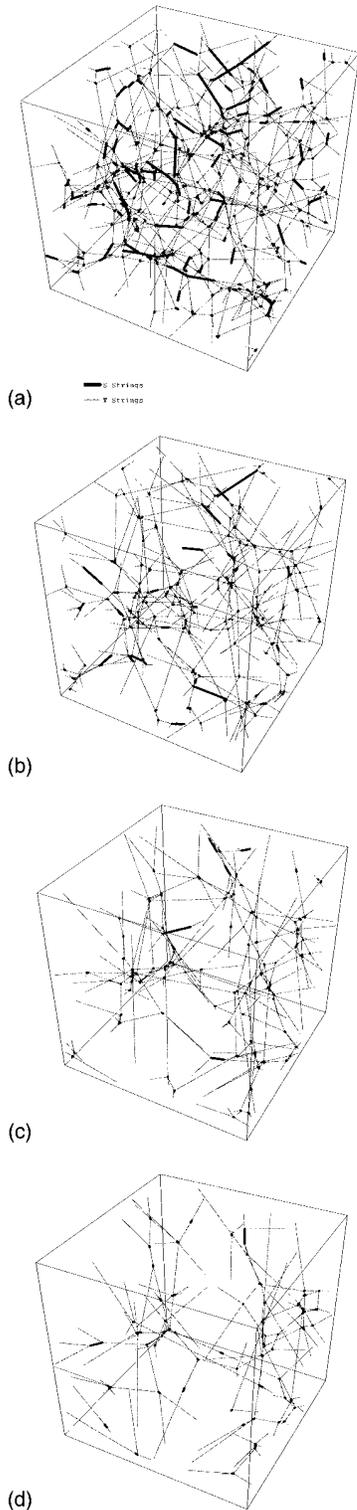
(c)

(d)

FIG. 16. (a) Snapshots from the evolution of an S_3 network with heavy S strings, from lattice gauge initial conditions. The S strings, or even strings, are shown in thicker lines. The full 10^3 simulation volume is shown. Notice the rapid shrinking of even segments, which leads to the formation of clusters that are slow to untangle. The first frame shown here is at $t=1.5$. (b) $t=3.0$. (c) $t=4.5$. (d) $t=6.0$.

shown for comparison. Figures 15–19 show three-dimensional pictures of simulated cosmic string networks during their evolution [31]. The pictures show several of the different patterns of evolution that can occur under different
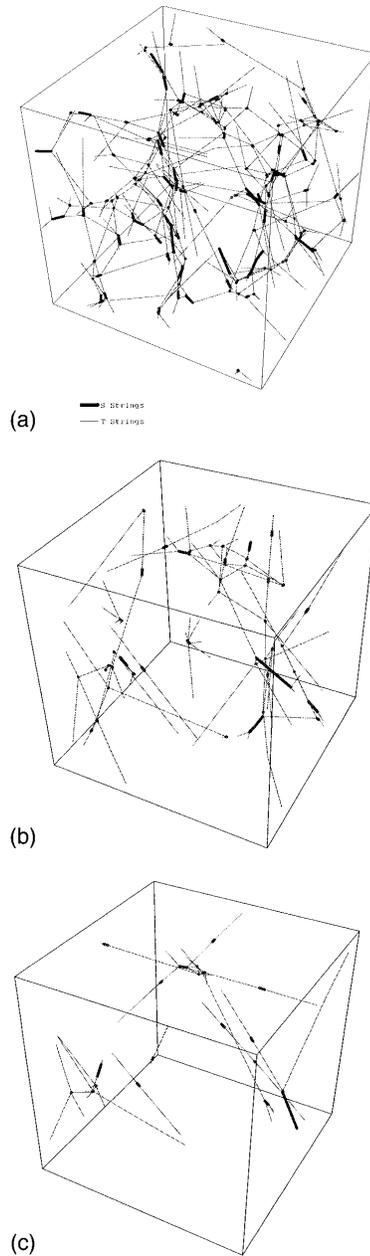

(a)

(b)

(c)

FIG. 17. (a) Evolution of a network from Higgs initial conditions with heavy S strings. The qualitative behavior is rather similar to that of the denser lattice gauge network, except that the network disappears much more rapidly, with large voids opening up very quickly. This figure: $t=1.0$. (b) $t=2.0$. (c) $t=3.0$.

conditions. Pictures of a Z_3 network are provided as well. The plots in Figs. 20–22 show the evolution of some length-scale variables as functions of time. These variables include D_n, the inverse cube root of the density of nodes, the average segment length between junctions for each type of string, and the two quantities D_s and D_t. The last two are analogous to D_l defined above; the inverse square root of the string length per unit volume, computed separately for each type of string. (Some of these variables may be redundant.) With some exceptions discussed below, these length scale variables exhibit (after some transients at early times) the linear increase characteristic of self-similar evolution. Such plots were made for simulations run under a wide variety of
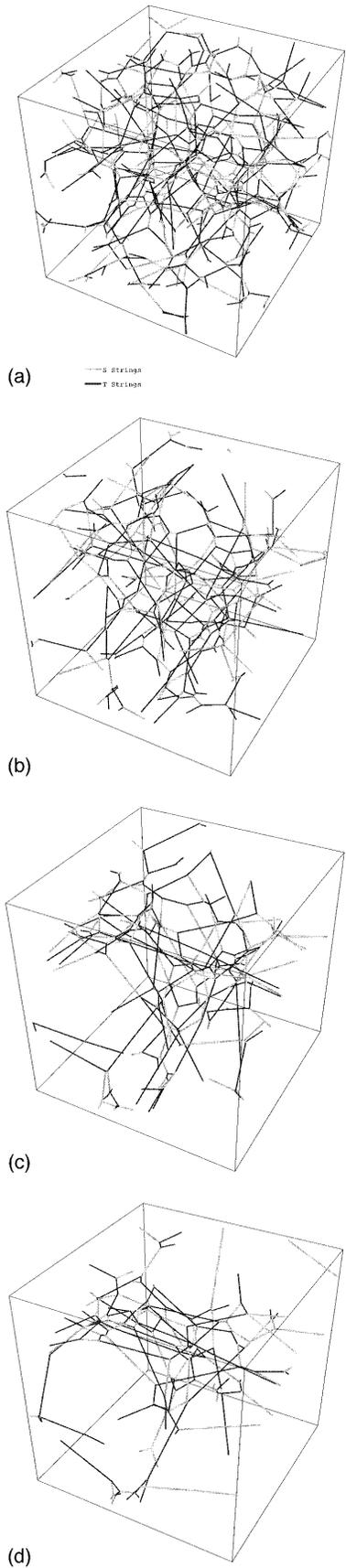
FIG. 18. Evolution of a network with equal string tensions, from lattice gauge initial conditions. (a) $t=2.0$. (b) $t=4.0$. (c) $t=6.0$. (d) $t=8.0$.
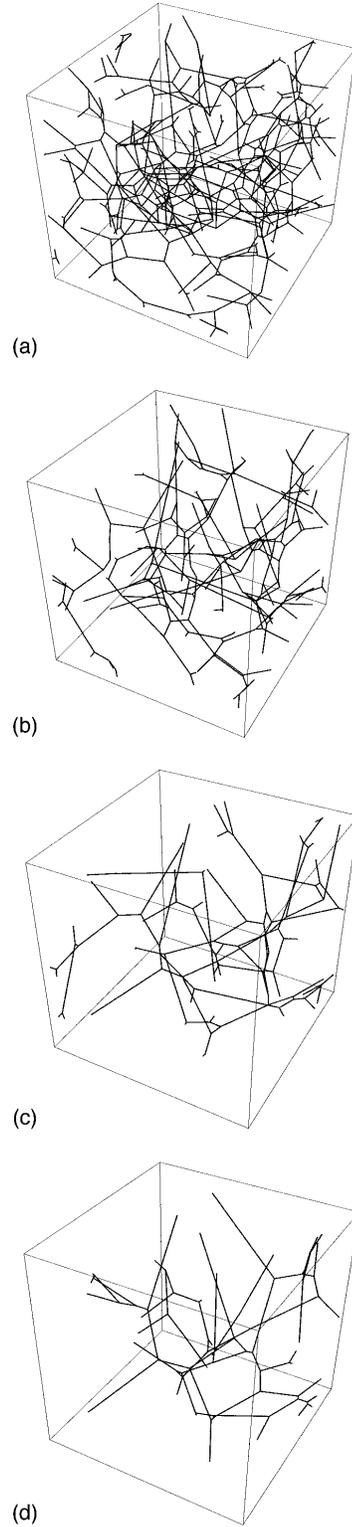

FIG. 19. Snapshots from the evolution of a pure Z_3 network, included for comparison with the non-Abelian network. (a) $t=0$. (b) $t=2.0$. (c) $t=4.0$. (d) $t=6.0$.

conditions giving a survey of the simulation's parameter space. A few representative plots are shown here in order to show their typical shapes. The remainder are summarized in Table II, which gives their slopes. The last set of figures in this section, Figs. 23–25, consists of a series of histograms which show how the distribution of string segment lengths
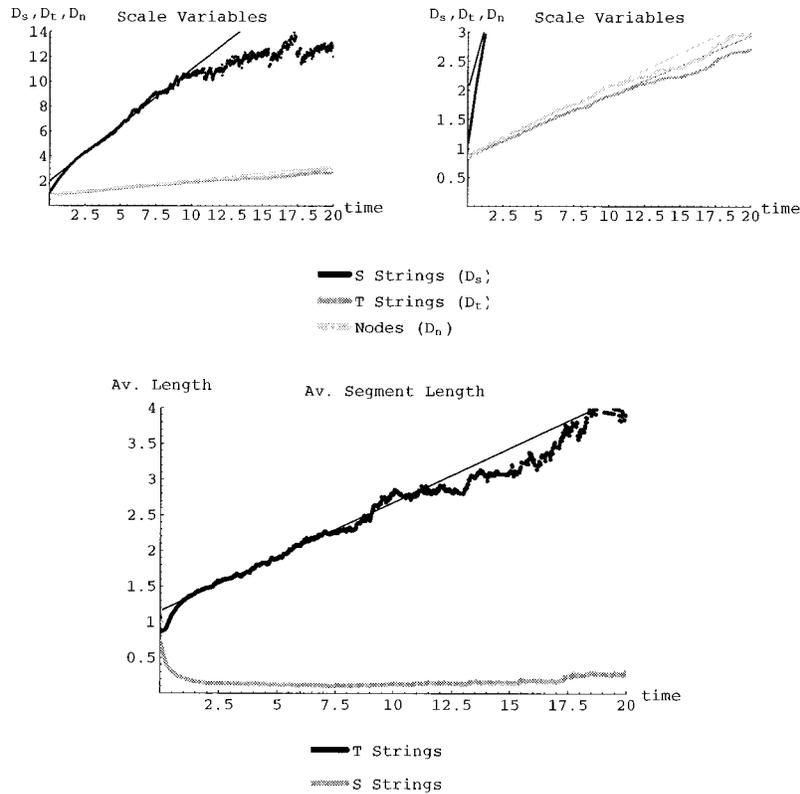
FIG. 20. Scaling-law variables as a function of time for S_3 network with $T_s = 2$, lattice gauge initial conditions, bridge NCC and no rearrangement. The top two plots are the curves of D_s, D_t, and D_n shown at two different magnifications. All three variables are plotted on the same axes, color coded as indicated. The bottom plot shows the average segment length for each of the two string types.

evolves with time. Distributions are shown for two contrasting cases discussed below, and corresponding data is also provided for the Z_3 network for the sake of comparison.

The parameter space surveyed in Table II includes two different initial condition simulations, several different ratios of the string tensions, and two other binary choices affecting string collisions and close encounters between nodes. The choice between ''bridge'' and ''zipper'' configurations for colliding non-Abelian strings is one choice (abbreviated B and Z in Table II). The other choice determines what happens when two neighboring nodes are within distance d_{ann} but are topologically unable to annihilate. In the cases labeled R, such nodes undergo a rearrangement of connections (see previous section) with a probability of 0.2 per time step. In the cases labeled N, no rearrangements are allowed, and the nodes simply bounce.

As a rough measure of the importance of intercommutations and NCC's, Table II also gives, in the last two columns, the ratio of the total number of NCC events to the number of intercommutations, and the ratio of the number of NCC's to the cumulative net number of nodes annihilated. (For the Z_3 network, there are no NCC's and the number of intercommutations is shown instead.)

Unless otherwise specified, all numerical quantities are given in units such that the initial lattice spacing is 1, the displacement of each node during a time step is given by $\Delta x = \Sigma_r T_r \Delta t$, and the magnitude of the tension T_t of the *odd* strings is normalized to 1. (This normalization was chosen because the special status of odd strings: they are the ones

which cannot end, and in none of the cases simulated did they show a tendency to become extinct. Even strings, on the other hand, disappear almost completely under certain conditions.) The tension T_s of the even strings was varied relative to this fixed value.

One salient feature of the results is hardly unexpected: the ratio of the tensions of different string species has a strong effect on the behavior of the network. The case of heavy s strings, in particular, is an exceptional one which will be discussed further below. Much more surprising is the strong difference in behavior between networks with different initial conditions. Evidently, different initial conditions lead to quite different trajectories which appear self-similar.

We will address the issues in the following order. First, we discuss the different evolution patterns that occur with different choices of string tension, focusing on the contrasting limits of heavy and light even strings. Then we comment on other effects, including the effects of different initial conditions. Finally, at the very end of the section, we will make a few remarks about uncertainties and finite-size effects.

S_3 network with light even strings. Consider a network in which the even, or three-cycle, strings have a much lower tension than the odd strings. In this case, the odd strings may pass through each other with comparative ease by forming new segments of the lighter even string. The odd strings may shrink with comparatively little energy cost in the creation of new strings. Furthermore, a zipper-type collision of two odd strings may have a result which, from the point of view of the odd string subsystem, resembles an intercommutation
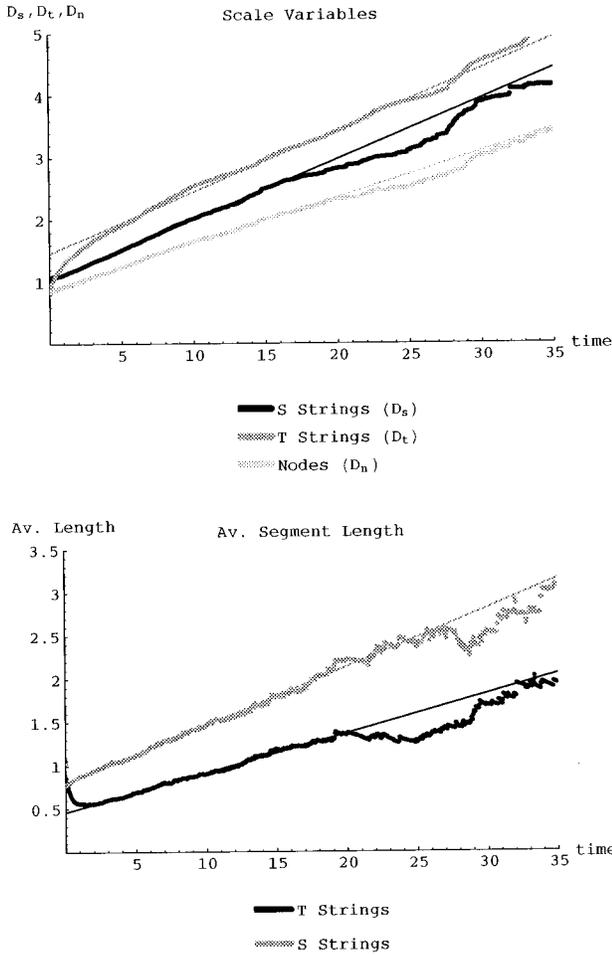
FIG. 21. Length scales for S_3 network with light S strings. T_s $=0.5$, lattice gauge initial conditions, bridge NCC, no rearrangements.

(see Fig. 26). The newly formed light string offers comparatively little resistance to the straightening of the rejoined heavy strings. We might expect that in this limit, the odd strings might behave as a network of Z_2-like (i.e., unbranched and unoriented) strings moving through a viscous medium formed by the branched Z_3 network of even strings. The odd strings can shorten and cut themselves into decaying loops while transferring part of their energy to the even string network. The even network, in turn, can dissipate its energy in much the same way as a Z_3 network.

Simulation results seem consistent with the above picture. Figure 15 shows a series of portraits of the evolution of a network with tensions $T_s=0.5$, $T_t=1.0$, beginning from lattice gauge initial conditions. The configuration at time $t=0$ is in fact the one shown shown in Fig. 12. In the early stages of evolution, we see a large number of highly crumpled, apparently Brownian t strings with thick webs of s string stretched between them. As time progresses, the odd strings begin to straighten and some closed loops shrink away and vanish. However, even though the odd strings are shortening at the expense of stretching the even ones, the population of even strings is at the same time being reduced through the annihilation of nodes, with the result that neither species of
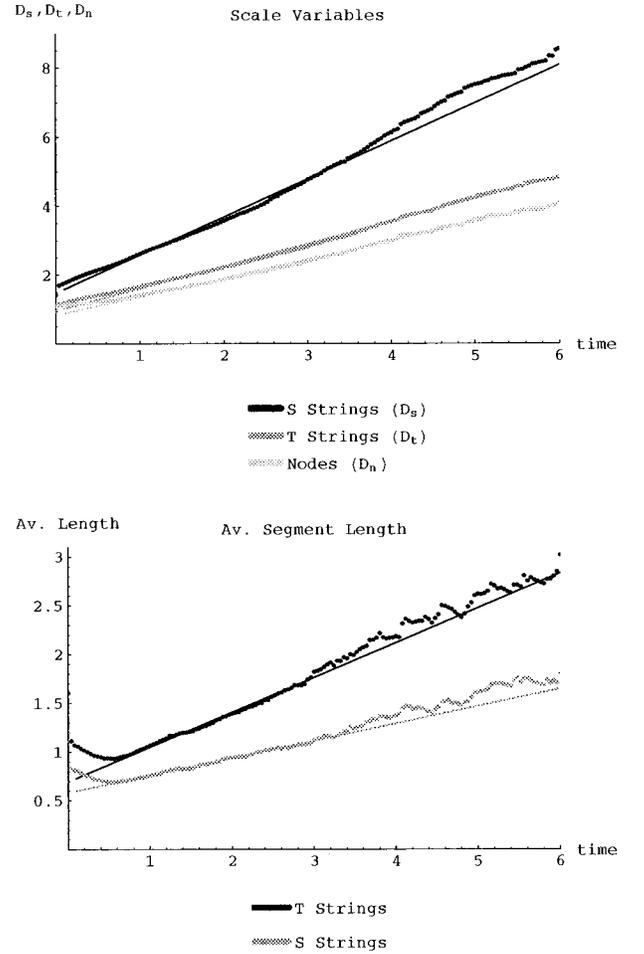
FIG. 22. Length scales for S_3 network with equal string tensions, Higgs initial conditions, bridge NCC, rearrangement allowed.

string becomes entirely dominant over the other. Instead, the evolution appears self-similar, with an approximately constant ratio between the amounts of the two different string species. This constant ratio is evident in Fig. 21 from the linear evolution of both length scale variables D_s and D_t. Apparently, after some transient behavior at very early times, a sort of dynamical equilibrium is established, with energy being transferred in a steady cascade from the t network to s network and then lost to damping. Interestingly, decreasing the even string tension still further to 0.25 does not allow the odd strings to contract more quickly (see Table II)—evidently they are impeded by the larger population of even strings.

S_3 *network with heavy even strings.* A situation which contrasts with the case of light s strings is one in which the s strings have a large tension. In this subsection, we discuss simulation results for the case with $T_s=2$, $T_t=1$. A glance at Figs. 16 and 20 reveals that this leads to quite different results. The s, or even, strings are in this case only marginally stable against decay into pairs of t strings. If a heavy s string ends on a t string, then the two t segments pulling against the s string at the junction cannot prevent the s segment from shrinking unless the angle between the t segments is sufficiently small; in this marginal case of $T_s=2$ they must be collinear in order for the tensions to balance. In either lattice
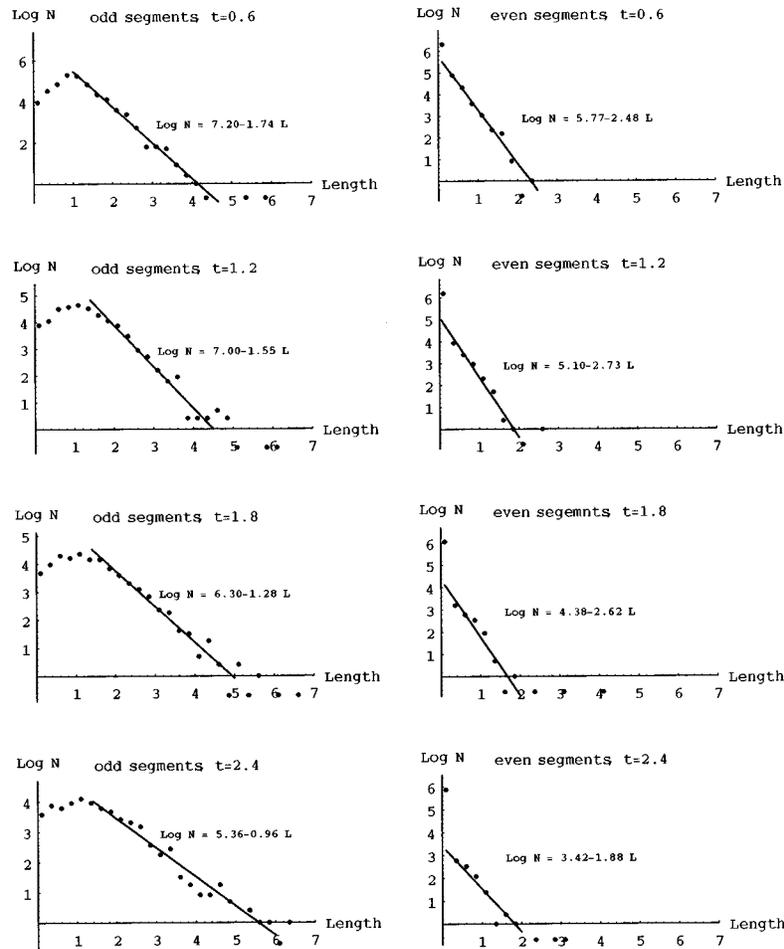
FIG. 23. Evolution of the distribution of length of even (S) and odd (T) string segments between junctions, for a network with heavy S strings ($T_S = 2$).

gauge or discrete Higgs initial networks, the majority of s strings end on t strings, and so there is very little to prevent these heavy strings from shrinking rapidly.

The shrinking of an individual s segment frequently brings together a pair of nodes which are topologically unable to annihilate. The nodes stick together because to separate them again would require stretching the s segment. Similar adhesion happens whenever two noncommuting t strings collide. Soon the network consists predominantly of odd strings stretched between small tangled clusters, and no further annihilation can occur until separate clusters meet, or the geometry changes sufficiently to allow some strings to pull free. Figure 20 shows the scaling variables as functions of time. The rapid disappearance of s string is evident from the extremely fast increase in the variable D_s in the upper plot, and from the lower plot which shows that the average length of an s segment quickly drops to near zero. This is an exceptional case: in all other cases besides $T_s = 2$, the average segment length for both string types increases with time.

Even though the s strings do not obey the usual standard scaling behavior, it appears that the average t segment length does increase linearly with time, as do D_n and D_t. Evidently, the overall system obeys scaling once most of the s strings have shrunk.

The prevalence of clusters connected by short s segments is apparent in the snapshots of Fig. 16 and is illustrated in a

different manner by Fig. 23, which shows semilogarithmic histograms of the distribution of segment lengths at a sequence of times. The length distributions for the two string types evolve in different ways. For lengths larger than one lattice spacing, the t segments follow an exponential distribution (which appears linear on semilogarithmic axes) with a steadily increasing decay length. While the few long s segments that remain are also exponentially distributed, there is no clear tendency for the decay length of this distribution to increase with time, and the number of long segments quickly shrinks into insignificance compared with the large population of very short segments in the lowest bin of the histogram. The presence of so many short segments is associated with the adhesion of many pairs of nodes.

Compare Fig. 23 with the length distributions in the $T_s = 0.5$ case, shown in Fig. 24. In the latter, we see that both distributions develop a peak at short lengths (indicating that some clusters do occur), but the peaks are less pronounced and the number of such short segments decreases at a rate commensurate with the remaining population of strings. The exponential decay lengths for both distributions increase with time, as expected in a self-similar evolution. Evidently, the pile-up of clusters is less significant in the light-s-string case. Length distributions for the Abelian network (Fig. 25) show no excess at all at short distances.

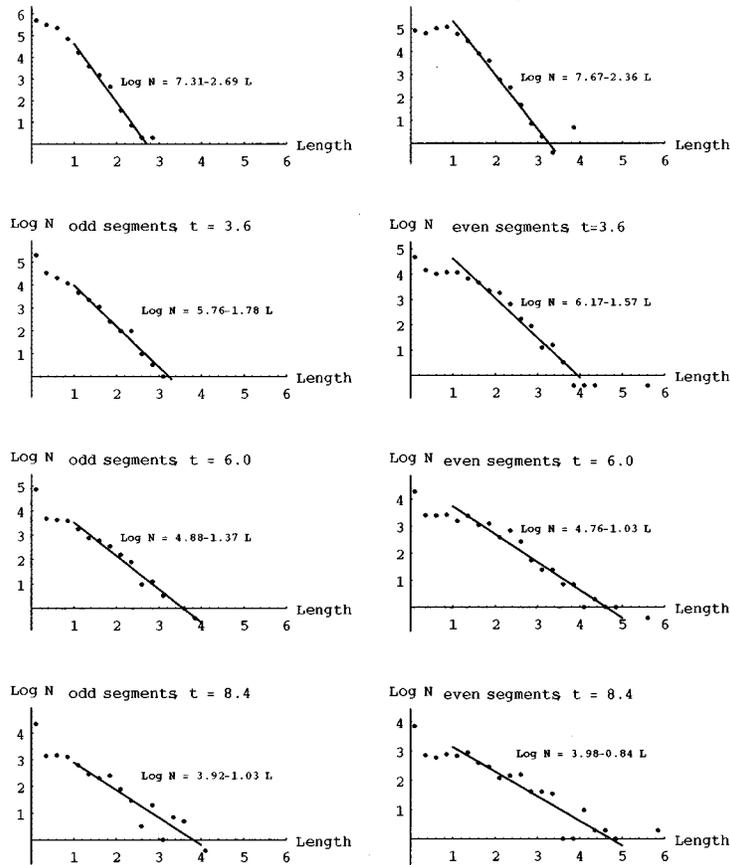*Other effects.* The non-Abelian simulation has been run

FIG. 24. Segment length distributions for network with light S strings ($T_S = 0.5$).

under a variety of conditions. For the most part, the results are consistent with some type of self-similar evolution. Variables having the dimensions of length all increase linearly with time. The exception is for networks with heavy s strings, in which the average s segment length decreases. This is the only case for which one type of string dominates completely over the other. In all other cases, the amounts of the two string types approach a constant ratio.

The evolution of the network has some interesting dependence on parameters other than the string tensions. Most surprising is the very strong dependence on the statistics of the initial conditions. When the Higgs initial conditions are used, the network decays much faster, especially in the case of heavy s strings. Not only the absolute sizes, but the ratios of coefficients can have quite different values depending on which initial conditions are used. This dependence is somewhat mysterious. The lattice gauge initial conditions are much more dense than the Higgs initial conditions, but the
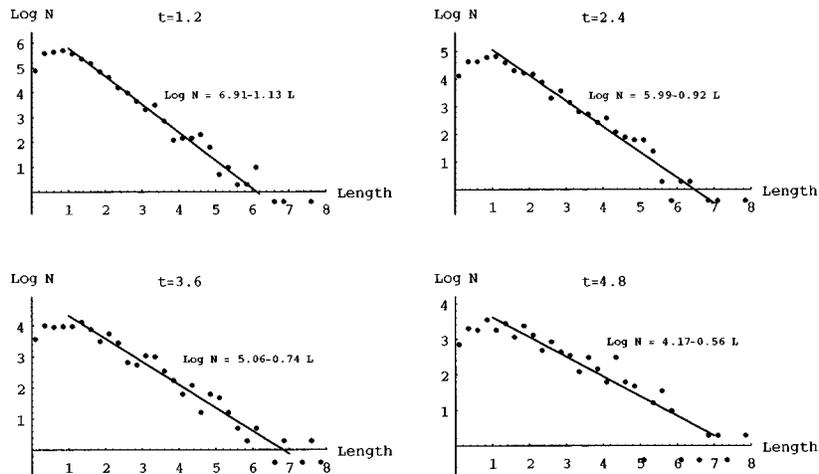
FIG. 25. Segment length distributions for the Abelian Z_3 network, included here for comparison with Figs. 23 and 24.

TABLE II. Coefficients for network's evolution: rates of change for D_s, D_t, D_n, and average segment length of each type, ratio of NCC's to intercommutation events, and cumulative number of NCC's per node, reported for different conditions. The final row gives Z_3 results. Uncertainties (estimated by analyzing subsets of the data) are approximately ± 2 in the last decimal place, or ± 0.02 for most of the values given. Abbreviations: LG=lattice gauge initial conditions, H=Higgs initial conditions, B=bridge NCC, Z=zipper NCC, R=rearrangements allowed, N=no rearrangements, IC=intercommutation, NCC=noncommutative collision.

Conditions	dD_s/dt	dD_t/dt	dD_n/dt	$d(\text{avg}.S)/dt$	$d(\text{avg}.T)/dt$	NCC/IC	NCC/node
$T_s=2.0$, LG							
B N	0.9	0.10	0.11	<0	0.15	5.9	0.83
Z N	0.7	0.10	0.11	<0	0.15	8.6	0.93
B R	1.4	0.25	0.40	<0	0.37	5.6	0.64
Z R	1.5	0.29	0.35	<0	0.63	8.1	0.91
$T_s=2.0$, H							
B N	2.7	0.55	0.69	<0	1.2	4.2	0.23
Z N	2.8	0.46	0.59	<0	1.1	4.5	0.25
B R	3.0	0.71	0.91	<0	2.0	2.9	0.17
Z R	3.7	0.82	1.1	<0	1.9	4.0	0.22
$T_s=1.0$, LG							
B N	0.15	0.15	0.11	0.08	0.09	5.3	0.33
Z N	0.16	0.12	0.10	0.05	0.08	5.8	0.23
B R	0.30	0.20	0.18	0.11	0.17	6.3	0.06
Z R	0.34	0.22	0.17	0.09	0.11	3.8	0.06
$T_s=1.0$, H							
B N	1.0	0.56	0.50	0.19	0.41	3.9	0.17
Z N	1.0	0.53	0.43	0.12	0.33	3.0	0.10
B R	1.1	0.65	0.54	0.18	0.35	2.8	0.05
Z R	1.2	0.72	0.61	0.21	0.33	3.1	0.04
$T_s=0.5$, LG							
B N	0.10	0.10	0.07	0.05	2.9	2.9	0.37
Z N	0.09	0.13	0.07	0.03	0.03	2.4	0.23
B R	0.17	0.20	0.13	0.07	0.07	2.5	0.09
Z R	0.17	0.26	0.13	0.07	0.07	2.3	0.07
$T_s=0.5$, H							
B N	0.19	0.15	0.10	0.04	0.05	2.1	0.17
Z N	0.20	0.20	0.13	0.07	0.11	6.3	0.02
B R	0.47	0.43	0.30	0.18	0.20	2.5	0.06
Z R	0.44	0.39	0.29	0.16	0.17	2.0	0.05
$T_s=0.5$, LG							
Z R	0.09	0.17	0.08	0.04	0.06	1.9	0.13
$T_s=1$, Z_3 (with intercommutations)	0.37		0.38	0.25			0.12

overall density of nodes decreases during the evolution, soon reaching a density comparable to that of the initial Higgs-initial-condition-generated network, yet the two continue evolving at very different rates. Evidently, some statistical property other than the total density is important, such as, for example, the distribution of voids, or even correlations of the fluxes. A better understanding of this phenomenon should be the goal of future work.

Notice that the details of the behavior of closely approaching nodes makes some difference in the evolution. The inability of some pairs of nodes to annihilate is apparently an important impediment to the network's destruction. Rearrangements (quasi-intercommutations) of the connec-

tions of neighboring nodes evidently increase the mobility of flux, and increase the likelihood that eventually some neighboring pairs of nodes will be able to annihilate. As is apparent from Table II, the inclusion of rearrangements almost invariably speeds up the decay of the network, often by roughly a factor of 2. The choice of either zipper or bridge configurations for colliding noncommuting strings generally makes a smaller difference, if any.

Finite-size effects, fluctuations, and uncertainties. A few remarks are in order concerning the interpretation of results and the effects of simulating on a finite volume. The size of our simulation volume is limited in practice by computation time; the flux computations (see Appendix) are computation-
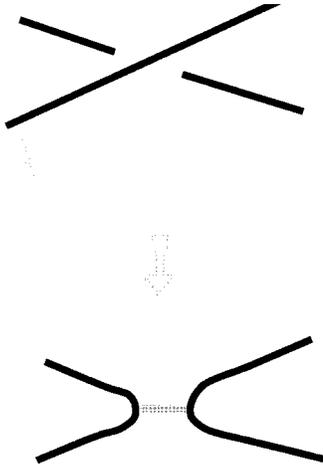
FIG. 26. Two colliding t strings form an s-string zipper. If the s string is ignored, this event looks very similar to an intercommutation.

ally intensive. Typical plots of scale variables like those in Figs. 20–22 exhibit some transient behavior at early times, followed by a period of linear increase. As the typical distance between nodes increases, however, a point is eventually reached where only a few nodes remain within the simulation volume. At this point, both systematic and random deviations from linear scale growth begin to occur. With fewer nodes, curves become bumpy due to lower statistics. Many strings wind all the way around cycles of the periodic boundary, so that the simulated network can no longer be expected to mimic an infinite one. A typical run meets one of two distinct fates at the end of its period of self-similar evolution. It may undergo a very sudden collapse as the last remaining nodes annihilate, leaving either no strings at all or a few strings stretched across the entire volume. This sudden collapse is visible as a sharp upward turn in the plots of length scale versus time. Alternatively, the network may reach a stable or metastable configuration with a small number of nodes (typically of order 10 or fewer.) The characteristic length scale ceases its linear increase and reaches a plateau. Needless to say, we cannot make reliable inferences about an infinite network once the scale of our simulation volume becomes the important one, and we can only extract information on scaling behavior from the linear part of the curve. The conclusion that our data are consistent with scaling is based on the existence of a linear stretch which typically continues until at least one of the length scale variables has grown to one half the width of the simulation volume. We cannot rule out slow (e.g., logarithmic) deviations from scaling.

With respect to fluctuations, not all simulation runs are alike. Runs with lattice gauge initial conditions and $T_s = 2$ seemed to undergo quite a bumpy evolution characterized by periods of temporary freezing alternating with cascades of annihilation. The bumpiness was smoothed out only by averaging over multiple runs. Runs with $T_s = 2$ without rearrangement almost always end with a few nodes remaining rather than with complete collapse. This is reflected in Fig. 20, where the curves of D_t and D_n versus time begin leveling off as the average segment length approaches 4 (which is half the width of the volume). This leveling is probably a

finite-size effect and not indicative of the behavior of an infinite network for the following reasons: It occurs when only a few nodes are left in the simulation volume, and it occurs at later times as the simulation volume is increased. In fact, one run performed on a larger 10^3 volume did not level off at all, but collapsed entirely instead. Higgs-initial-condition-generated networks tend to evolve more smoothly and to be more likely to undergo total collapse at the end of the simulation.

VI. CONCLUSIONS AND DISCUSSION

One of the motivations for undertaking a simulation of non-Abelian string dynamics was to test for deviations from the familiar power-law behavior of the network's energy density as a function of time, and especially to look for evidence for or against the conjecture that the tangling of strings would cause a non-Abelian network to freeze into a static (fixed in comoving coordinates) equilibrium. The results we have found suggest that such a scenario, if it can occur at all, requires very special conditions. Over a range of different regimes, the results found here are consistent with some form of self-similar evolution, and with a density decaying at a rate commensurate with that of matter, $n = Ct^{-2}$. The coefficient C, however, can vary in quite interesting ways due to non-Abelian effects. In a cosmological scenario, this coefficient controls the fractional contribution of strings to the energy of the universe.

The interplay between the different string species causes the self-similar evolution to be realized in some novel ways which can be quite different under different conditions. Features such as the ratio of the populations of different string species may depend in rather complicated ways on various factors. Particularly striking and new is the strong influence of the initial distribution of strings on the subsequent evolution. Evidently, different self-similar evolutionary trajectories are possible, and initial conditions may be attracted to one trajectory or another depending on some statistical feature other than the network's overall density. This behavior seems almost paradoxical, going against the notion that a scaling evolution is one which has no memory of its initial state.

There are interesting questions to investigate at both the ''microscopic'' and ''macroscopic'' level. By microscopic questions, we mean those concerning the dynamics of individual strings and individual collisions. Our results indicate that the behavior of nodes in the network as they encounter each other has a controlling influence on the network's evolution. The inability of non-Abelian nodes to annihilate is an important impediment to the removal of strings, and configurations with nodes close together or coalescing into composites appear to be especially important in the interesting case of heavy s strings. The process we have called ''rearrangement'' of two nearby nodes seems to increase the mobility of strings and allow the network to decay more quickly. This simulation was run with rather *ad hoc* assumptions about node collisions. A better microscopic understanding of nodes' behavior and that of multinode tangles will help provide input for improved simulations and understanding of macroscopic questions.

At the macroscopic level, it would be desirable to obtain a

better understanding of the principles governing the interplay between different string types and the strong influence of initial conditions. The types of network behavior seen here may open up new possibilities for cosmological model building, aside from the string domination scenario. Some of the types of structures seen here, such as webs of light strings stretched between heavier ones (as in the light-s-string scenario of this simulation) or the tangled clusters seen in the $T_s = 2$ simulation might have interesting cosmological effects. We have seen hints that the strength of fluctuations in string density is different under different circumstances, being especially strong for $T_s = 2$. Might these fluctuations be interesting sources of structure formation? Along with analytical study, more refined simulations of branched and non-Abelian networks may well prove worthwhile.

APPENDIX: SIMULATION OF NON-ABELIAN FLUXES

The subtle nature of non-Abelian magnetic flux in discrete gauge theories has been the subject of quite interesting research [15,16,20,32,33], and presented algorithmic and computational challenges for this simulation. Ambiguities in the definitions of non-Abelian fluxes, mentioned in Sec. II, require that we design a very careful procedure to fix consistent definitions and maintain their consistency as the strings move. This must be done in order to allow us to make the appropriate comparisons of fluxes when two strings collide or two nodes attempt to annihilate. Additional subtleties occur as a result of our using periodic boundary conditions: we are dealing with a discrete gauge theory on a nonsimply connected space manifold, and holonomies associated with nontrivial cycles become important.

This appendix describes our procedure for defining and comparing the fluxes of non-Abelian strings. Further details may be found in [18]. As in Sec. II, we use the formalism developed in [16,20].

Gauge fixing conventions. In our algorithm, the strings and nodes exist inside a rectangular volume with opposite sides identified: a three-torus. The subtleties associated with the periodic boundary conditions will be discussed later: for now we simply consider a network inside a rectangular volume with boundaries. We choose a cubic volume with one corner at $(0,0,0)$ and with side length L. As explained in Sec. II, it is necessary to define fluxes using paths that begin and end at some base point. We choose a base point x_0 at the center of our simulation volume, $x_0 = (L/2, L/2, L/2)$. Let each node be associated with a straight line segment (a "tail") along the direction $\overrightarrow{BN}$ from the base point to the node's location. Then let the flux of each outgoing string be defined with respect to a path which runs outward along this
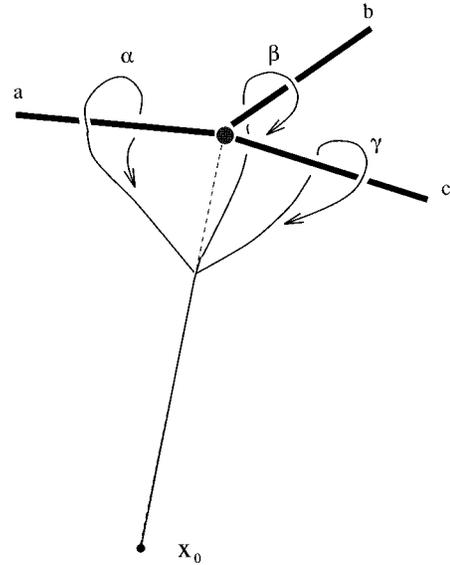

FIG. 27. Conventions for measuring the fluxes of the three strings emanating from a node. Each string's flux is defined as the flux through a path which leaves the base point x_0 along a straight line toward the node, then encircles the string in a counterclockwise direction as seen from the far end of the string and returns to the base point.

tail to a point which is taken to be vanishingly close to the the node. The path then encircles the string in a counterclockwise direction and returns to the base point along the node's tail. This is illustrated in Fig. 27. This will be our convention for defining the fluxes of the strings which join at a given junction.

As illustrated in Fig. 28, flux conservation requires that the product of all three fluxes emanating from a node be trivial when the fluxes are multiplied in a clockwise order with respect to the direction $\overrightarrow{BN}$. That is, if the strings in clockwise order are a, b, and c, then

$$cba = e. \tag{A1}$$

In our algorithm, a record is maintained of the geometry of each node: the strings carry labels indicating the appropriate clockwise orientation.

In the case of a doubly linked pair of nodes (Fig. 4) two segments are collinear, and the order is therefore ill defined. In such a case we allow the order to be arbitrary, but the fluxes of the two strings must be defined in a way consistent with that order, such that the product of all three fluxes is as usual trivial. The ordering must also be compatible between the two nodes which the segments join, so that the flux of a given segment is consistent at its two ends. (The consistency of segments from one end to another will be discussed below.)

The collection of standard paths defined above represents a set of generators for $\pi_1(\mathcal{M} - \{D\})$. The flux state of a network of strings is fully specified when we know the fluxes enclosed by all of these standard paths. The condition (A1) supplies one set of relations among these generators. For each string segment, there is also a relation involving the fluxes defined at its two end points, as discussed below.
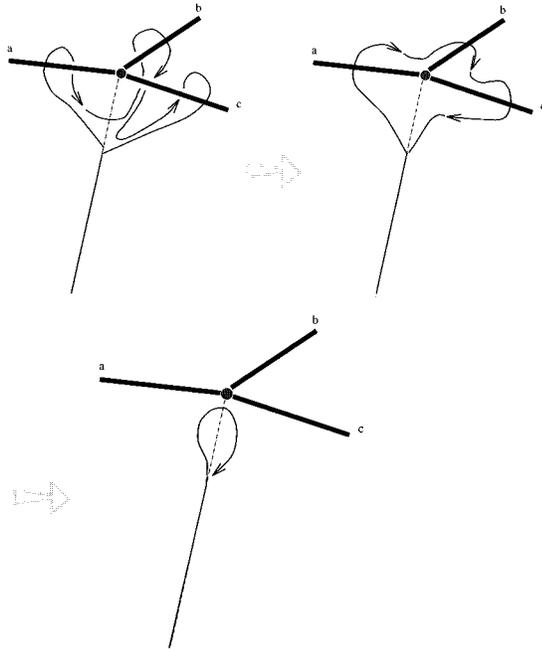
FIG. 28. The composition $\gamma\beta\alpha$ of all three paths can be continuously deformed to a point. Therefore cba, the product of all three fluxes taken in a clockwise direction as seen from above the node, must be trivial.
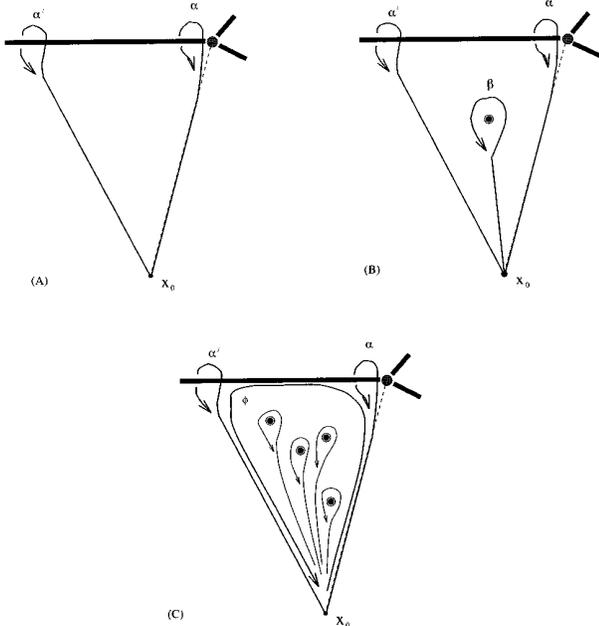

FIG. 29. When the flux a of a string has been defined according to a path which encircles it near one end, the flux a' of that string at another point along its length can be defined by "sliding" the standard path α to α' as shown. If no other strings pierce the triangle which is swept out, then this merely represents a continuous deformation of α, and thus $a' = a$ (A). However, if the triangle is pierced by string with flux b as measured by path β, then the flux is conjugated by b: $a' = bab^{-1}$ (B). More generally, if the triangle [or the oriented path ϕ shown in (C)] encloses flux f, then a is conjugated by the total flux f, i.e., $a' = faf^{-1}$. The total flux is given by the product of individual string fluxes, taken in order of increasing angle from the initial tail. (This can be seen by deforming a product of loops to a single loop enclosing all strings.)
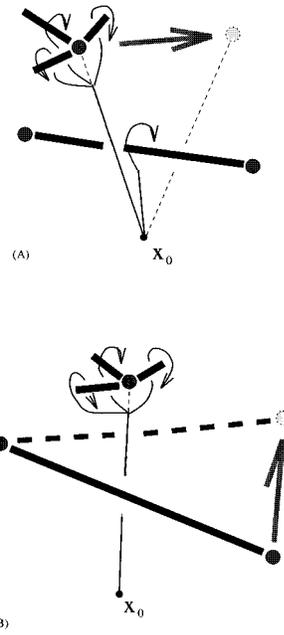

FIG. 30. Crossing of a node's tail by a string. This can happen either when the moving node drags its tail across the string (A), or when the string is dragged across the tail due to the motion of another node (B). In both cases, the fluxes of all strings attached to the node whose tail is crossed must be conjugated by the flux of the crossing string.

"Sliding" flux from the end point. By the conventions above, the flux of each string is defined at its two end points. But for the purposes of this simulation it is necessary to make comparisons of the fluxes of strings at arbitrary points along their lengths. For example, if two strings cross each other, their fluxes must be compared at the crossing point in order to determine whether they commute. A meaningful comparison of the fluxes of nearby string segments can be obtained only if the paths used to define those two fluxes remain close to each other everywhere except in the immediate vicinity of the strings to be compared. In particular, the "tails" of the paths must not pass on opposite sides of any string, because such paths would give different flux measurements for the same string. It is possible to define the flux of a string at an arbitrary point along its length by sliding the standard path to the one which encircles the string at the point we wish to measure, as illustrated in Fig. 29. If another string with flux b pierces the triangle which is swept out by the sliding path, then the flux at the new position is conjugated by b. If multiple strings occur, then the new flux a' is given by faf^{-1}, where f, the total flux inside the triangle, is defined as the product of the fluxes of all enclosed strings, taken in order of increasing angle from the initial ray $\overrightarrow{BP}$. The flux of each other string at the point where it pierces the triangle must in turn be defined by a similar sliding procedure from one of its ends. This procedure, applied recursively, can thus define the flux of any string at an arbitrary point P along its length, as measured by a path which follows a straight line from x_0 towards P and encircles the string near P.

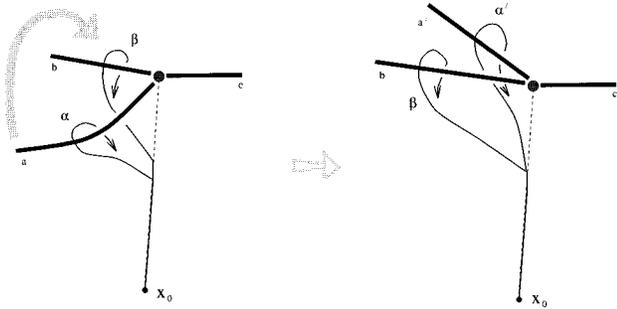If one slides the path all the way to the far end of the

FIG. 31. Example of a holonomy interaction between strings attached to the same node. When the string carrying flux a is lifted over the other string carrying flux b, its flux must be redefined as $a' = bab^{-1}$, and the conventional clockwise order of the three strings changes, with a and b exchanging places. The flux conservation condition is maintained: if $cba = e$ originally, then also $ca'b = e$.
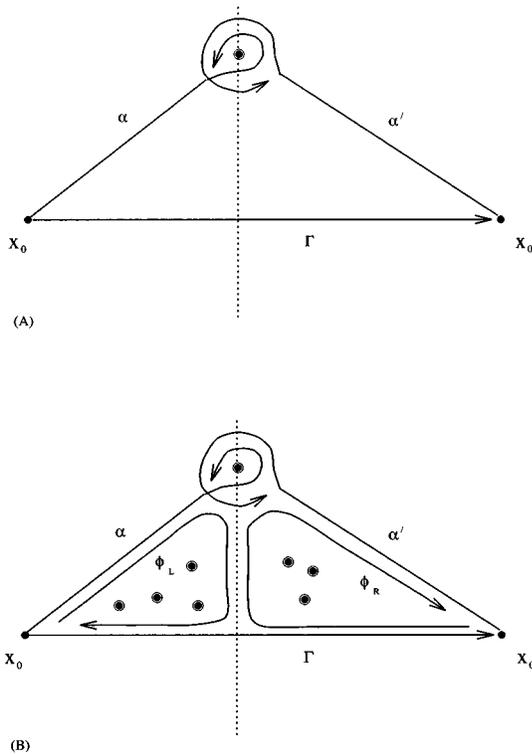

FIG. 32. Transformation from one description of a flux to another at the boundary. Here a string is shown intersecting the plane of the page precisely where it intersects the boundary of the cubic simulation volume (dotted line). Under periodic boundary conditions, the two points labeled x_0 are identified. The flux of the string may be described in terms of a path whose tail extends to the right of x_0 (α) or to the left (α'). If no other strings are present, then α is homotopically equivalent to $\Gamma^{-1} \alpha' \Gamma$. In the more general situation shown in (B), $\alpha \sim (\phi_L \Gamma \phi_R)^{-1} \alpha' (\phi_L \Gamma \phi_R)$, and so the two descriptions of the flux are related through conjugation by $f_L C f_R$, where C is the flux associated with the path Γ and f_L and f_R are the overall fluxes enclosed by ϕ_L and ϕ_R, respectively. The latter can be defined in terms of paths lying entirely on one side or the other of the boundary.
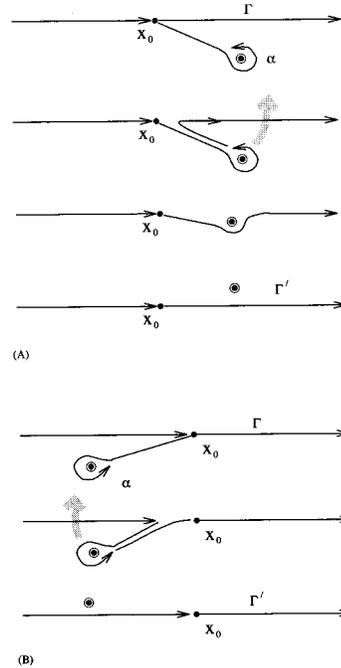

FIG. 33. Interaction between a string and one of the large loops of the three-torus. As a string with flux a as defined by the path α crosses to the right of x_0 as shown in (A), the path $\alpha\Gamma$, where Γ is a straight-line path which wraps around the three-torus, can be continuously deformed to the new straight-line path Γ'. Thus the flux C associated with Γ must be multiplied from the right by the string's flux, $C' = Ca$. If the string crosses to the left as in (B), $\alpha\Gamma$ is deformed to Γ' and so the multiplication is from the other side: $C' = aC$.

string, the resulting value of the flux must be consistent with the value measured by the standard path at the other end. This specifies an additional set of relations among the generators of $\pi_1(\mathcal{M} - \{D\})$ and furnishes one way of testing for errors in the simulation, as well as being necessary in order to define the fluxes of strings at the newly created nodes that result from string collisions.

A modified version of this sliding procedure is used to define all fluxes initially from the original lattice. Paths composed of lattice links are deformed by a series of steps to straight-line paths from the base point to the location of each node.

Holonomy interactions. As the network evolves dynamically and nodes change their position, the fluxes defined by these conventions may change in several different ways. First, as a node moves, its tail may be dragged across another string segment. Conversely, a string segment may be dragged across the node's tail by the motion of other nodes. In both cases, the fluxes of all strings at the node must be conjugated by the flux which is crossed, as shown in Fig. 30. In addition, the geometry of the strings at a given junction may change, resulting in holonomy interactions among the three strings joined at that node. Such a process is shown in Fig. 31: The motion of string a causes its standard flux to change, and also changes the clockwise ordering of the strings a, b, and c. This requires both an adjustment of the flux definitions and of the order labels.

Periodic boundary conditions. Because of the periodic boundary conditions, our simulation volume has the topology of a three-torus T_3. The nontrivial topology introduces three additional classes of noncontractible closed loops other than the ones associated with strings, namely, those which wrap around one of the boundaries. These loops may be associated with nontrivial flux. As in the case of vortices on a Riemann surface [34], the fluxes of these loops may have holonomy interactions with the fluxes of strings, and therefore a full description of string fluxes requires us to maintain a record of these large loops.

As representatives of the three ''wrap around'' classes, we choose canonical straight-line paths parallel to the coordinate axes, which we will refer to as Γ_x, Γ_y, and Γ_z. Γ_x, for instance, leaves the base point along the $+\hat{x}$ direction, wraps around the boundary from $x=L$ to $x=0$, and then returns to the base point from the $-\hat{x}$ side. Along with values of the fluxes for all strings, our algorithm maintains a record of the S_3 holonomies C_x, C_y, and C_z associated with Γ_x, Γ_y, and Γ_z, respectively. These values must be known in order to make consistent comparisons of string fluxes across the boundary. (The procedure for doing so is illustrated by Fig. 32.) The values of the C_i may change if a string crosses one of the curves Γ_i (see Fig. 33).

---

[1] T. W. B. Kibble, Phys. Rep. **67**, 183 (1980); J. Phys. A **9**, 1387 (1976).

[2] Ya. B. Zel'dovich, I. Yu. Kobzarev, and L. B. Okun, Zh. Eksp. Teor. Fiz. **67**, 3 (1974) [Sov. Phys. JETP **40**, 1 (1975)].

[3] J. Preskill, Phys. Rev. Lett. **43**, 1365 (1979).

[4] For a review, see, A. Vilenkin, Phys. Rep. **121**, 263 (1985); or more recently M. B. Hindmarsh and T. W. B. Kibble, Rep. Prog. Phys. **58**, 477 (1995).

[5] Ya. B. Zel'dovich, Mon. Not. R. Astron. Soc. **192**, 663 (1980); A. Vilenkin and Q. Shafi, Phys. Rev. Lett. **51**, 1716 (1983).

[6] A. Vilenkin, Phys. Rev. Lett. **46**, 1169 (1979); **46**, 1496(E) (1979).

[7] A. Vilenkin, Phys. Rev. Lett. **53**, 1016 (1984).

[8] D. Spergel and U. L. Pen, astro-ph/9611198.

[9] H. B. Nielsen and P. Oleson, Nucl. Phys. **B61**, 45 (1973).

[10] T. Vachaspati and A. Vilenkin, Phys. Rev. D **30**, 2036 (1984); M. Hindmarsh and K. Strobl, Nucl. Phys. **B437**, 471 (1995).

[11] E. P. S. Shellard, in *Cosmic Strings: the Current Status*, edited by F. S. Accetta and L. M. Krauss (World Scientific, Singapore, 1988), pp. 25–31.

[12] E. P. S. Shellard, Nucl. Phys. **B283**, 264 (1988); K. Moriarty, E. Myers, and C. Rebbi, Phys. Lett. B **207**, 411 (1988); in *Cosmic Strings: the Current Status* [11], pp. 11–24; R. Matzner and J. McCracken, *ibid.*, pp. 32–41.

[13] A. Albrecht and N. Turok, Phys. Rev. Lett. **54**, 1868 (1985); Phys. Rev. D **40**, 973 (1989); D. P. Bennet and F. R. Bouchet, Phys. Rev. Lett. **60**, 257 (1988); **63**, 2776 (1989); Phys. Rev. D **41**, 2408 (1990); B. Allen and A. P. S. Shellard, Phys. Rev. Lett. **64**, 119 (1990); B. Allen and A. P. S. Shellard, in *The Formation and Evolution of Cosmic Strings*, edited by G. Gibbons, S. Hawking, and T. Vachaspati (Cambridge University Press, Cambridge, England, 1990), pp. 421–448.

[14] T. Vachaspati and A. Vilenkin, Phys. Rev. D **35**, 1131 (1986).

[15] F. Bais, Nucl. Phys. **B170** [FS1], 32 (1980).

[16] M. Alford, K. M. Lee, J. March-Russell, and J. Preskill, Nucl. Phys. **B384**, 251 (1992), and references therein.

[17] G. Toulouse, J. Phys. Lett. (Paris) **38**, L-67 (1976); V. Poenaru and G. Toulouse, J. Phys. (Paris) **38**, 887 (1977).

[18] P. McGraw, Report No. CALT-68-20044, hep-th/9603153; Ph. D. thesis, Caltech, 1996.

[19] M. Alford *et al.*, Phys. Rev. Lett. **64**, 1632 (1990); **65**, 668(E) (1990); Nucl. Phys. **B349**, 414 (1991).

[20] M. Bucher, Nucl. Phys. **B350**, 163 (1991).

[21] For more discussion of homotopy groups and their application to topological defects, see, M. Nakahara, *Geometry, Topology, and Physics* (Adam Hilger, New York, 1990); N. D. Mermin, Rev. Mod. Phys. **51**, 591 (1979).

[22] I. N. Herstein, *Abstract Algebra* (Macmillan, New York, 1986).

[23] M. Hindmarsh and T. W. B. Kibble, Phys. Rev. D **55**, 2398 (1985).

[24] M. Daniel, G. Lazarides, and Q. Shafi, Nucl. Phys. **B170**, 156 (1979).

[25] J. Preskill, Annu. Rev. Nucl. Part. Sci. **34**, 461 (1984).

[26] L. M. A. Bettencourt, P. Laguna, and R. A. Matzner, Phys. Rev. Lett. **78**, 2066 (1997); L. M. A. Bettencourt and T. W. B. Kibble, Phys. Lett. B **332**, 297 (1994).

[27] T. Vachaspati and A. Vilenkin, Phys. Rev. D **30**, 2036 (1986); T.W.B. Kibble, Phys. Lett. **166B**, 311 (1986).

[28] M. Aryal, A. E. Everett, A. Vilenkin, and T. Vachaspati, Phys. Rev. D **34**, 434 (1986).

[29] W. M. Fairbairn, T. Fulton, and W. H. Klink, J. Math. Phys. **5**, 1038 (1964).

[30] The Z_3 lattice gauge algorithm and the statistics of its initial network were also described in the work of L. Nayvelt and J.E. Woods, report, Caltech, 1988 (unpublished).

[31] Animations are available through the World Wide Web at URL http://physics.unc.edu/~mcgraw/stringmovies.html

[32] H. K. Lo, Phys. Rev. D **52**, 7247 (1995); hep-th/9502079; hep-th/9411133.

[33] M. Alford, S. Coleman, and J. March-Russell, Nucl. Phys. **B351**, 735 (1991); M. Alford, J. March-Russell, and F. Wilczek, *ibid.* **B337**, 695 (1990).

[34] K. M. Lee, Phys. Rev. D **49**, 2030 (1994).