Neural, Parallel, and Scientific Computations 17 (2009) 385-408

CONSTRAINED MULTIOBJECTIVE CONTROL PROBLEMS: APPLICATION TO SOCIAL NETWORKS

N. G. MEDHIN AND G. L. PORTER

Department of Mathematics, North Carolina State University, Raleigh, NC 27695

ABSTRACT. Social networks involve studying how relations form between individuals in a group based on their shared preferences and attributes. This research addresses a very difficult question involving how social networks arise and evolve over time. Historically, some researchers have addressed this issue using loglinear modeling, continuous time Markov theory or rational choice theory. In this work, social force theory is used to model social interaction and overall network dynamics while multiobjective control theory provides a basis for predicting network structural formation. Using computer simulations, we numerically analyze the evolution and long-term behavior of optimal network structures based on the demographics of a small dataset. We pay special attention to the effect that memory has on friendship choices and clique formation.

Key words: Multiobjective control, social forces, actor, clique, friendship, sociomatrix, preferences, attributes

1. INTRODUCTION

Social network analysis focuses on relationships among social entities and the implications from such relationships. In recent decades, social network analysis has received a lot of attention from social and behavioral scientists, who leverage it to answer standard research questions about politics, economics and society at large [9]. Apparently, being able to organize separate entities into networks and groups is very important in determining social and economic outcomes. For instance, personal contacts provide valuable job information; networks are important to “trade and exchange of goods in non-centralized markets”; and finally, organizing societies into groups allows the proper distribution of public good and services. The applications do not stop there; the literature on social networks is vast and covers a variety of topics with recent applications even extending to national defense.

In Section 2, we formulate the multiobjective control problem, and provide a numerical algorithm which generates a set of Pareto-optimal solutions to the problem. In Section 3, we define key terminology for studying social networks followed by a detailed explanation of the social forces model for network dynamics. The model explains social interaction and basis for friendship choice. Section 4 provides a computer

Received July 13, 2009

simulation in which the optimal solution of multiobjective control problem is used to form a social matrix; the resulting analysis shows that our social forces model yields realistic network behavior. In addition, Section 5 adds a memory effect to the social forces model to show the impact that long-term memory has on actors' friendship choices. Finally, we explore the model's capability to answer questions concerning clique connection and network destabilization in Section 6.

2. MULTIOBJECTIVE OPTIMAL CONTROL

The goal when solving multiobjective optimal control problems (MOCPs) is to optimize a set of conflicting objectives simultaneously while satisfying constraints on the system under consideration. MOCPs have garnered great attention from researchers in recent years since many real-world problems tend to possess multiple, conflicting objectives which must be optimized yet there seems to be no single accepted definition for "optimum" in this case as with the single objective optimal control problem. We describe the multiobjective control problem in detail as follows.

2.1. **Problem Formulation.** The general form of the control problem without memory is (see Section 2.2)

$$\min_{\mathbf{u}\in U} \mathbf{J} = [J_1(\mathbf{u}), \dots, J_s(\mathbf{u})]^T$$

such that

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{x}(t_0) &= \mathbf{x}^0 \end{aligned}$$

and

$$\begin{aligned} h_i(\mathbf{x}, \mathbf{u}) &= 0 \quad \text{(equality constraints)}, \quad i = 1, \dots, r \\ h_i(\mathbf{x}, \mathbf{u}) &\leq 0 \quad \text{(inequality constraints)}, \quad i = r+1, \dots, k \end{aligned}$$

where

$$J_l(\mathbf{x}^0, \mathbf{u}(t), \mathbf{x}(t)) = \Phi_l(\mathbf{x}(t_f)) + \int_0^{t_f} L_l[\mathbf{x}(t), \mathbf{u}(t)]dt, \qquad l = 1, \dots, s$$

2.2. **Optimality Condition.** When there are multiple and conflicting objectives, usually there is no one solution that minimizes all of them at once. Therefore, we must define what we mean by "optimal" for the multiobjective control problem. A solution $u^* \in U$ is a **Pareto-optimal** solution [5], [6] to the MOCP if there does not exist any other $u \in U$ for which

1. $J_i(\mathbf{u}) \leq J_i(\mathbf{u}^*)$ for all i and
2. $J_j(\mathbf{u}) < J_j(\mathbf{u}^*)$ for at least one j.

Pareto-optimal solutions are also referred to as *noninferior*, *nondominated* or *efficient* solutions. Pareto-optimal solutions are not unique. Instead, we get a whole set of alternative solutions, one no better than the other.

2.3. **Algorithm: Differential Evolution.** Differential Evolution (DE) is a population-based search method created by Storn and Price [8] to handle problems with multiple objectives over continuous domains. DE is an appealing approach for solving MOCPs because it eliminates the need to consider function continuity, convexity, or concavity unlike some traditional search techniques where such complexities must be given great attention. In addition, DE is capable of providing a complete set of Pareto-optimal solutions in a single run [7].

2.3.1. *Steps for Differential Evolution (DE) Algorithm.*

- **Step 1:** Random Population Initialization
  In this step, $\mathbf{u}^g_{j,i}$ means the i-th entry of the vector $\mathbf{u}^g_j$. We initialize the population as follows:

$$\mathbf{u}^g_{j,i} = \mathbf{u}^g_{j,i_{min}} + rand() * (\mathbf{u}^g_{j,i_{max}} - \mathbf{u}^g_{j,i_{min}}), \quad j = 1, 2, \dots, NP,$$

  g is the current generation and $rand()$ is a random number in $[0, 1)$. The i-th component of the vector $\mathbf{u}^g_j$, $j = 1, 2, \dots, NP$, has a lower bound, $\mathbf{u}^g_{j,i_{min}}$, and an upper bound, $\mathbf{u}^g_{j,i_{max}}$.
- **Step 2:** Mutation
  For each $j = 1, 2, \dots, NP$, pick $j_1, j_2, j_3 \in \{1, 2, \dots, NP\}$ randomly and form the vector $\hat{\mathbf{z}}^g_j$ according to the formula:

$$\hat{\mathbf{z}}^g_j = \mathbf{u}^g_{j_1} + W * (\mathbf{u}^g_{j_2} - \mathbf{u}^g_{j_3}), \qquad j = 1, 2, \dots, NP$$

  where j_1, j_2, j_3 are mutually different and not equal to j. The parameter W is a scaling factor for mutation and is usually a value between 0 and 1.
- **Step 3:** Crossover
  As in Step 1, we denote the i-th component of the vector z^g_j by $z^g_{j,i}$. The operation *crossover* is implemented as follows:

$$z^g_{j,i} = \begin{cases} u^g_{j_1,i} + W * (u^g_{j_2,i} - u^g_{j_3,i}) & \text{if } rand() < CR \text{ or } i = \hat{i}, \\ u^g_{j,i} & \text{otherwise} \end{cases}$$

  where $\hat{i}$ is a randomly selected index from $\{1, 2, \dots, D\}$.
- **Step 4:** Selection

$$\mathbf{u}^{g+1}_j = \begin{cases} \mathbf{z}^g_j & \text{if } J(\mathbf{z}^g_j) \le J(\mathbf{u}^g_j), \\ \mathbf{u}^g_j & \text{otherwise} \end{cases}$$

- **Step 5:** Termination criteria in the literature often includes running the algorithm for some maximum number of generations or until some desired objective function value is reached.

3. SOCIAL NETWORKS

Several key concepts [9] form the basis of social network analysis and are fundamental to our study of social networks.

3.1. **Methodology for Social Networks.** *Nodes* form the basis of social networks and are often referred to as actors, actors or points depending on the context of discussion. Nodes in a social network can be social entities such as people, businesses, organizations, cities, nations, etc. An *edge* is a line connecting nodes. Edges are also referred to as links, ties, lines or arcs, representing a relationship or connection between a pair of nodes. In network analysis, there are many types of ties to include behavioral interaction ties (i.e., conversing or emailing), physical movement ties (i.e., migration) and individual evaluation ties (i.e., friendship among actors which is the focus of this paper). Network ties are often made based on some type of individual or entity attributes. *Attributes* describe characteristics of actors in a group. For example, for a friendship network, such attribute variables might include income potential, gender, race, sex, education level, political tendency, religious affiliation, marital status, etc. In fact, measurements on actors' attributes often constitute the make-up of social data and social networks.

There are two tools in particular which are often seen in the literature to represent social networks: *matrices* and *graphs*. In this work, we'll use both in illustrative examples of friendship networks. A *sociomatrix* is the primary matrix used in social network analysis and is denoted by $\mathbf{X}$. If there are N actors in a social group, then the sociomatrix for the group would be an $N \times N$ matrix of binary entries representing the relations between the actors. Each actors in the sociomatrix has a row and column both indexed $1, 2, \ldots, N$. The entries in the sociomatrix, x_{ij}, represent which nodes are linked. For our friendship model, relations in the sociomatrix may be directional and nondirectional which will lead to both symmetric and nonsymmetric sociomatrices. For symmetric sociomatrices, if two actors are friends, there will be a 1 in the ij-th and ji-th cells and a 0 if they're not friends. The ii-th cells will contain a value of 0 since actors do not befriend themselves. For nonsymmetric sociomatrices, while the ij-th cells may contain a 1, this may not be the case for the ji-th cell if the relation is not reciprocated.

A *graph* (often referred to as *digraph*) has a set of nodes representing the actors in the network and a set of lines to represent the existence of ties or links between pairs of actors. The graph can be drawn directly from the sociomatrix. Since relations in our model may or may not be symmetric, lines are both directional and nondirectional. In essence, if a directional line exists from actor i to j, it may not exist from j to i. We exclude any loops, which are lines between actors and themselves since actors do not befriend themselves.

3.2. **Social Forces Model for Social Networks.** Different modeling approaches have been developed to model social networks and social interaction. In this work, we take a more physical approach inspired by Helbing's social forces model for pedestrian walking behavior. We adapt Helbing's model to describe social interaction and ultimately, formulate a friendship model mathematically using the notion of social forces. In essence, actors interact as though they were subject to acceleration and repulsive forces when making their friendship choices. This approach assumes that individuals behave according to a set of rules in a manner that promotes their utility minimization, i.e, they choose courses of action with the most benefit and least cost. In the context of friendship networks, social forces theory assumes that each actor possesses a specific attitude toward making friends, a desire to befriend those who share their preferences and attributes and that they respect the private space of others. Consequently, following Helbing and Molnar's theory, these rules describing social interaction can be placed into a set of equations of motion [4].

3.2.1. *Assumptions.* We start with a fixed set of actors, denoted Λ, consisting of N actors, who begin as mutual strangers and enter into social relationships with other actors as time evolves. We make the following assumptions [3] in our model of network dynamics:

- All actors consider the same attributes when attempting to make friends.
- Actors do not change categories within a particular attribute.
- Relationships between actors depend on shared preferences for attributes and categories.
- Reciprocity for numerical preference levels is automatic by virtue of using the Euclidean distance as a measurement of closeness but this is not so for categorical preferences.
- Each actor attempts to maximize his status in the social group, i.e, he wishes to form as many relationships as possible.
- Finally, the objective functional of each actor decreases with an increase in shared attribute preferences and categories.

3.2.2. *Data.* The following data is required to run our model of network dynamics:

Data:

N – total number of actors in a social environment

m – total number of attributes under consideration

l – total number of categorical attributes under consideration

k – number of categories in a particular categorical attribute

$\mathbf{r}_i(t)$ – position vector describing actor i's preference for each attribute, $1, \ldots, m$

$\mathbf{y}_i$ – vector identifying various attribute categories to which actor i belongs

$\mathbf{w}_i$ – vector containing actor i's preferences for similar attribute categories

$\mathbf{v}_i^0$ – vector describing actor i's initial rate of change of attribute preferences at time $t = 0$

$\mathbf{v}_i(t)$ – vector describing actor i's rate of change of attribute preferences at time t

$\mathbf{u}_i(t)$ – vector describing actor i's control for each attribute, $1, \ldots, m$

Parameters:

l_{ij} – constant value set to ensure that actor j respects the private space of actor i

τ_i – relaxation time taken by each actor to return to his $\mathbf{v}_i^0$

$\mathcal{N}_i$ – reflects an actor's desire to stick to his belief system

Now that we have formally stated what each data variable represents, we can describe a few variables in more detail. For instance, $\mathbf{v}_i^0$ is meant to reflect how quickly a person intends to change their preference on a certain attribute in order to make friends; it is represented by a "velocity" vector in the social forces model described in Section 3.3 and hereafter, we will call it intended *social velocity*. Therefore, if a person intends to change their attribute preference levels rapidly, we'd expect to see a larger $\mathbf{v}_i^0$ compared to those who intend to change less rapidly. Similarly, $\mathbf{u}_i(t)$ controls how much actors vary their attribute preferences within a given set of bounds in order to make friends. The control variables of people who desire to make many friends will fluctuate greatly when compared to those actors who desire fewer relationships, reflected by control variables which are greatly restricted. Similarly, since l_{ij} controls how close actors allow others to get to them, those actors who desire to make many friends will have a larger value for l_{ij} than those who desire to keep others at a distance. Further, a large $\mathcal{N}_i$ is meant to penalize an actor for deviating from his belief system and thus results in an increase in an actor's performance index. Finally, τ_i will be small for those who are more reluctant to change their attribute preferences permanently.

3.3. **The Model.** It is well documented that individuals tend to behave in ways that maximize a utility function of interest. Thus, we can formulate a multiobjective optimal control problem (MOCP) involving a set actors who wish to make as many individual relations as possible while minimizing the associated costs and maintaining their core beliefs. We use the optimal solution of the MOCP to form the social matrix of the social group. The problem can be stated as follows.

3.3.1. *Problem Formation.*

$$\min_{\mathbf{u}} \sum_{j\neq i} \|\mathbf{r}_i(t_f) - \mathbf{r}_j(t_f)\|^2 + \sum_{j\neq i} \left\|\mathbf{w}_i(t_f)\cdot(\mathbf{y}_i(t_f) - \mathbf{y}_j(t_f))\right\|^2 + \mathcal{N}_i \int_{t_0}^{t_f} \|\mathbf{u}_i(t)\|^2\, dt \tag{1a}$$

such that

$$\dot{\mathbf{r}}_i = \mathbf{v}_i \tag{1b}$$

$$\begin{aligned}\dot{\mathbf{v}}_i = {} & \frac{1}{\tau_i}(\mathbf{v}_i^0 - \mathbf{v}_i) - \nabla_{\mathbf{r}_i}\sum_{j\neq i}\|\mathbf{u}_i - \mathbf{u}_j\|^2 \\ & \cdot(1 + ((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - \mathbf{v}_j\Delta t\|)^2 - \|\mathbf{v}_j\Delta t\|^2) \\ & \cdot\exp\{-(l_{ij}((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - v_j\Delta t\|)^2 - \|\mathbf{v}_j\Delta t\|^2))\}\end{aligned} \tag{1c}$$

and

$$(\mathbf{r}_i(0) - \vec{\delta}_{i_{min}}) \leq \mathbf{r}_i(t) \leq (\mathbf{r}_i(0) + \vec{\delta}_{i_{max}}) \tag{1d}$$

$$\vec{\delta}_{i_{min}} \leq \mathbf{u}_i(t) \leq \vec{\delta}_{i_{max}} \tag{1e}$$

Before stating exactly how to form the social matrix using the optimal solution, we describe the MOCP in detail.

3.3.2. *The Performance Index.* When forming friendships, the goal of each actor is to form as many ties as possible by minimizing the *social distance* between himself and others but not at the expense of his belief system. This desire is described in the following performance index (also referred to as the objective, cost, or payoff function):

$$\mathbf{J}_i = \sum_{j\neq i} \|\mathbf{r}_i(t_f) - \mathbf{r}_j(t_f)\|^2 + \sum_{j\neq i} \left\|\mathbf{w}_i(t_f)\cdot(\mathbf{y}_i(t_f) - \mathbf{y}_j(t_f))\right\|^2 + \mathcal{N}_i \int_{t_0}^{t_f} \|\mathbf{u}_i(t)\|^2\, dt.$$

We examine each term of the objective function separately. The first term

$$\sum_{j\neq i} \|\mathbf{r}_i(t_f) - \mathbf{r}_j(t_f)\|^2$$

is a component of social distance which represents the vector distance between actors i and j on their levels of preference for the various attributes under consideration.

The second term

$$\sum_{j\neq i} \left\| \mathbf{w}_i(t_f) \cdot (\mathbf{y}_i(t_f) - \mathbf{y}_j(t_f)) \right\|^2$$

is the final component of social distance; it represents the weighted categorical distance between actors i and j. The categorical distance, y_{ij}^k, between the two actors on attribute k is calculated as follows:

$$y_{ij}^k = \left| y_i^k - y_j^k \right| = \begin{cases} 1 & \text{if } y_i^k \neq y_j^k, \\ 0 & \text{if } y_i^k = y_j^k. \end{cases}$$

The vector $\mathbf{w}_i$ holds actor i's weighting factors for each categorical attribute. This weighting factor, $0 \leq w_i^k \leq 1$, describes the actor's attitude toward *similarity* on attributes when making friends. A larger w_i^k reflects that making friends with actors from similar attribute categories is of utmost importance to actor i while not as important for smaller w_i^k. Essentially, the weighting factor describes an actor's tolerance for diversity.

The third term

$$\mathcal{N}_i \int_{t_0}^{t_f} \left\| \mathbf{u}_i(t) \right\|^2 dt$$

represents the desire of each actor to stay as true to his beliefs as possible over time. $\mathcal{N}_i$ is a weight that actor i uses to express how strongly he desires to stick to his belief system. A large $\mathcal{N}_i$ reflects that actor i is less willing to deviate from his core values while a small $\mathcal{N}_i$ means that he does not care as much to stick to his beliefs .

3.3.3. *Social Network Dynamics.* People are very likely *familiar with* the situations they normally encounter. When reacting to issues that arise in their immediate environment, they usually choose the best decision based on past experience. Therefore, we can say that their reactions are somewhat *automatic* and predictable. This allows us to describe how they react or behave using the below set of equation of motions. Specifically, these equations form a system of nonlinear ordinary differential equations which describe the state dynamics and constraints for our multiobjective optimization problem.

Consider a set of actors $\Lambda = \{1, 2, \ldots, N\}$. Each actor, $i \in \Lambda$, is described by a position vector, denoted by $\mathbf{r}_i(t) = [r_i^1(t), r_i^2(t), \ldots, r_i^m(t)]^T$ and an associated *social velocity* vector, denoted by $\mathbf{v}_i(t) = [v_i^1(t), v_i^2(t), \ldots, v_i^m(t)]^T$. Each quantity, $\mathbf{r}_i^k$, in the position vector describes the actor's preference for particular attributes, k=$1, \ldots, m$ while $\mathbf{v}_i^k$ describes the intentions or motivation an actor has regarding making friends based on a particular attribute preference.

First, an actor changes his position according to the following differential equation:

$$\dot{\mathbf{r}}_i = \mathbf{v}_i.$$

Each actor's level of preference for the various attributes is allowed to fluctuate by some amount δ_i^k so that we have the following constraint on the position vector:

$$(\mathbf{r}_i(0) - \vec{\delta}_{i_{min}}) \leq \mathbf{r}_i(t) \leq (\mathbf{r}_i(0) + \vec{\delta}_{i_{max}})$$

Each actor has a vector of controls, $\mathbf{u}_i = [u_i^1, u_i^2, \ldots, u_i^m]^T$ used to vary his attribute preferences within the fixed attribute limits. The constraints on the control vector are represented as:

$$-\vec{\delta}_{i_{min}} \leq \mathbf{u}_i \leq \vec{\delta}_{i_{max}}.$$

Next, to describe the actor's change in *social velocity* over time, we use

$$\begin{aligned}
\dot{\mathbf{v}}_i &= \frac{1}{\tau_i}(\mathbf{v}_i^0 - \mathbf{v}_i) - \nabla_{\mathbf{r}_i} \sum_{j \neq i} \|\mathbf{u}_i - \mathbf{u}_j\|^2 \\
&\quad \cdot (1 + ((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - \mathbf{v}_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2)) \\
&\quad \cdot \exp\{-(l_{ij}((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - v_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2))\}.
\end{aligned}$$

This equation is of utmost importance to our model since it explains how each actor *moves* through time. Therefore, we explain each component of this term in careful detail.

The first term in the acceleration equation reflects an actor's desire to move as efficiently as possible and in a desired direction, $\mathbf{e}_i$, toward his next destination (or friend) with a certain speed or enthusiasm, $\mathbf{v}_i^0$. Yet, when an actor deviates from his intended *social velocity*, $\mathbf{v}_i^0$, within a specified relaxation time, τ_i, he has a strong tendency, $\mathbf{g}_i^0$, to approach his intended *social velocity* again. In practical terms, this means that having a larger τ_i will allow actors to deviate from their initial attitude toward changing attributes for longer periods of time in order to make many friends. Those actors with smaller τ_i values will be forced back to their initial attitudes quickly and will acquire fewer relationships. This effect is modeled by a *social velocity* term, g_i^0:

$$g_i^0(\mathbf{v}_i, \mathbf{v}_i^0) := \frac{1}{\tau_i}(\mathbf{v}_i^0 - \mathbf{v}_i).$$

The second term in the acceleration equation stems from the fact that the behavior of an actor, i, can be influenced by other actors, j, in his social group. While interacting with others, each actor generally respects the *private space* of other actors and tries not to get too close too fast. We can model these *territorial effects*, $\mathbf{g}_{ij}$, using a repulsive potential, $V_j(\beta)$:

$$\mathbf{g}_{ij}(\mathbf{r}_i - \mathbf{r}_j) = -\nabla_{\mathbf{r}_i} V_j[\beta(\mathbf{r}_i - \mathbf{r}_j)].$$

The interaction potential which is affected by each actor's behavior is defined by the sum of the repulsive potentials, V_j:

$$V_{int}(\mathbf{r}, t) := \sum_j V_j[\beta(\mathbf{r}_i - \mathbf{r}_j)]$$

where

$$\beta = (\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - \mathbf{v}_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2$$

As previously mentioned, actors require space to make their next move, which is respected by other actors. Therefore, for $V_j(\beta)$ we use a monotonic decreasing function in β. Since the previously mentioned effects or forces, all influence an actor's behavior simultaneously, their total effect equals the sum of all these forces. Therefore, the total motivation to act or the social force, $\mathbf{g}_i$ is:

$$\begin{aligned}
\mathbf{g}_i(t) &:= \mathbf{g}_i^0(\mathbf{v}_i, \mathbf{v}_i^0) + \sum_{j \neq i} \mathbf{g}_{ij}(\mathbf{r}_i - \mathbf{r}_j) \\
&= \frac{1}{\tau_i}(\mathbf{v}_i^0 - \mathbf{v}_i) - \nabla_{\mathbf{r}_i} V_{int}(\mathbf{r}_i, t)
\end{aligned}$$

where

$$\begin{aligned}
V_{int}(\mathbf{r}_i, t) = & \sum_{j \neq i} \|\mathbf{u}_i - \mathbf{u}_j\|^2 \left(1 + ((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - \mathbf{v}_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2)\right) \\
& \cdot \exp\{-(l_{ij}((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - v_j \Delta t\|)^2 - (\|\mathbf{v}_j \Delta t\|^2))\}
\end{aligned}$$

Hence, we have successfully described how an actor's attributes or preferences evolve through time when attempting to make friends and now we are able to state definitively that the equations 1 make up our social forces model for friendship dynamics [3].

3.4. **The Social Matrix.** We use the optimal solution from the MOCP described in the previous section to identify the group structure of a social network. First, we calculate the total social distance between two actors, i and j, by using

$$\begin{aligned}
\text{social distance} &= \text{numerical distance} + \text{weighted categorical distance} \\
d_{ij} &= \sum_{j \neq i} \|\mathbf{r}_i(t) - \mathbf{r}_j(t)\|^2 + \sum_{j \neq i} \left\|\mathbf{w}_i(t) \cdot (\mathbf{y}_i(t) - \mathbf{y}_j(t))\right\|^2
\end{aligned} \tag{2}$$

Before determining whether or not relationship ties exist using our model, we need to calculate the average social distance between the actors:

$$d_{avg} = \frac{\sum_{i=1}^{N} \sum_{j \neq i} d_{ij}}{N^2 - N}, \; j = 1, 2, \ldots, N. \tag{3}$$

Now, we are ready to determine whether or not two actors are connected. We decide the entries of the social matrix $\mathbf{X}$ using the rule

$$x_{ij} = \begin{cases} 1 & \text{if } d_{ij} \leq .8 d_{avg}, \\ 0 & \text{otherwise.} \end{cases}$$

So, if the social distance between two actors is less that the average social distance between actors, then we say that two actors are friends.

4. COMPUTER SIMULATION OF A SOCIAL NETWORK

Using the aforementioned performance index along with the nonlinear time varying dynamical system described by the social forces model from the previous section, we next provide an application to illustrate the features of the model more clearly as well as the usefulness of applying a multiobjective optimal control approach to study social networks.

In this example, we use our model to illustrate solving a multiobjective problem with numerous conflicting objectives. We'll show by computer simulation of interacting actors that our model is capable of producing realistic effects of friendship formation in a social network.

4.1. **Problem Statement.** Consider our model in Section 3.3.3 with $N = 25$. Specifically, we have the following objective functions to be minimized:

$$\begin{aligned}
J_1 = {} & \|\mathbf{r}_1(t_f) - \mathbf{r}_2(t_f)\|^2 + \|\mathbf{r}_1(t_f) - \mathbf{r}_3(t_f)\|^2 + \cdots + \|\mathbf{r}_1(t_f) - \mathbf{r}_{25}(t_f)\|^2 \\
& + \|\mathbf{w}_1 \cdot (\mathbf{y}_1(t_f) - \mathbf{y}_2(t_f))\|^2 + \|\mathbf{w}_1 \cdot (\mathbf{y}_1(t_f) - \mathbf{y}_3(t_f))\|^2 \\
& + \cdots + \|\mathbf{w}_1 \cdot (\mathbf{y}_1(t_f) - \mathbf{y}_{25}(t_f))\|^2 + \mathcal{N}_1 \int_{t_0}^{t_f} \|\mathbf{u}_1(t)\|^2 \, dt \\
J_2 = {} & \|\mathbf{r}_2(t_f) - \mathbf{r}_1(t_f)\|^2 + \|\mathbf{r}_2(t_f) - \mathbf{r}_3(t_f)\|^2 + \cdots + \|\mathbf{r}_2(t_f) - \mathbf{r}_{25}(t_f)\|^2 \\
& + \|\mathbf{w}_2 \cdot (\mathbf{y}_2(t_f) - \mathbf{y}_1(t_f))\|^2 + \|\mathbf{w}_2 \cdot (\mathbf{y}_2(t_f) - \mathbf{y}_3(t_f))\|^2 \\
& + \cdots + \|\mathbf{w}_2 \cdot (\mathbf{y}_2(t_f) - \mathbf{y}_{25}(t_f))\|^2 + \mathcal{N}_2 \int_{t_0}^{t_f} \|\mathbf{u}_2(t)\|^2 \, dt \\
& \vdots \\
J_{25} = {} & \|\mathbf{r}_{25}(t_f) - \mathbf{r}_1(t_f)\|^2 + \|\mathbf{r}_{25}(t_f) - \mathbf{r}_2(t_f)\|^2 + \cdots + \|\mathbf{r}_{25}(t_f) - \mathbf{r}_{24}(t_f)\|^2 \\
& + \|\mathbf{w}_{25} \cdot (\mathbf{y}_{25}(t_f) - \mathbf{y}_1(t_f))\|^2 + \|\mathbf{w}_{25} \cdot (\mathbf{y}_{25}(t_f) - \mathbf{y}_2(t_f))\|^2 \\
& + \cdots + \|\mathbf{w}_{25} \cdot (\mathbf{y}_{25}(t_f) - \mathbf{y}_{24}(t_f))\|^2 + \mathcal{N}_{25} \int_{t_0}^{t_f} \|\mathbf{u}_{25}(t)\|^2 \, dt
\end{aligned}$$

where $t_0 = 0$ and $t_f = 1$ which we try to minimize subject to the social forces model and constraints defined in Section 3.3.3.

4.2. **Data Sets Defined.** In order to simulate our model, we take $N = 25$ actors and $m = 5$ attributes (education level, age, income, political tendency, and religious affiliation). All of our attributes will serve as both quantitative and qualitative data in the sense that for each particular attribute, an actor must first state his level of preference for the attribute then describe the attribute category to which he belongs.

We construct a dataset to illustrate the capabilities of our model using a random sample, that is, we take 25 observations from a larger demographic dataset and derive numerical data needed for our model variables (see Tables 2, 3, 4 and 5). Specifically, the collection method involved poll responses to phone survey questions for the 2004 presidential election [1]. Questions covered demographics on education, age, income, political and religious preferences, etc. The overall demographics of the data set are presented in Table 1.

Table 1. Data Demographics

Attributes	**Categories**
Education Level	48% - Some College
	28% - Grad School
	24% - Grad School
Age Group	44% - 18 to 24
	36% - 25 to 36
	12% - 37 to 49
	8% - 50+
Income Level	52% - $< \$20K$
	28% - $20K to $40K
	20% - $50K to $70K
Political Affiliation	36% - Changes
	36% - Democrat
	20% - Republican
	8% - Other
Religious Affiliation	12% - None
	64% - Protestant
	12% - Other
	12% - Catholic

Challenges with data included missing data so to handle this, we chose only observations with complete data. Scale also presented a problem so we initialized data within the interval [0, 1] for comparison purposes.

TABLE 2. Model parameters for each actor: $i = 1, \ldots, 25$

Actor	l_{ij}	τ_i	N_i
1	0.05	1/15.0	1.0
2	0.05	1/15.0	1.0
3	0.15	1/10.0	1.0
4	0.25	1/10.0	1.0
5	0.25	1/5.0	1.0
6	0.25	1/5.0	1.0
7	0.25	1/5.0	1.0
8	0.05	1/15.0	1.0
9	0.05	1/15.0	1.0
10	0.05	1/15.0	1.0
11	0.15	1/15.0	1.0
12	0.25	1/5.0	1.0
13	0.15	1/10.0	1.0
14	0.25	1/5.0	1.0
15	0.25	1/5.0	1.0
16	0.05	1/15.0	1.0
17	0.05	1/15.0	1.0
18	0.25	1/5.0	1.0
19	0.15	1/10.0	1.0
20	0.05	1/15.0	1.0
21	0.25	1/5.0	1.0
22	0.15	1/10.0	1.0
23	0.15	1/10.0	1.0
24	0.15	1/10.0	1.0
25	0.25	1/5.0	1.0

4.3. **Implementation.** We use Differential Evolution (DE) to solve the multiobjective optimization problem:

$$\min_u \mathbf{J} = [J_1, J_2, \ldots, J_{25}] \tag{4}$$

subject to the constraints described in 3.3.3. Using this approach, we successfully generate a whole set of Pareto-optimal solutions, which are all equally good.

There were several computational challenges which have to be overcome to solve this problem. A system of 250 nonlinear ODEs must be solved and 25 nonlinear cost functions must be minimized. Using parameter recommendations for DE from [8] leads to a system that has approximately 12,500 parameters and a population size of

TABLE 3. Level of Preference, $\mathbf{r}_i$, for each attribute by actor: $i = 1, \ldots, 25$

Actor	Education	Age	Income	Politics	Religion
1	0.3915	0.5731	0.7367	0.7782	0.212
2	0.2664	0.4418	0.2583	0.3758	0.6072
3	0.7879	0.4397	0.6925	0.279	0.7069
4	0.3396	0.2403	0.3989	0.5451	0.2821
5	0.7207	0.3976	0.4928	0.4589	0.3563
6	0.7949	0.7931	0.5	0.7846	0.6355
7	0.5995	0.5	0.6016	0.5498	0.7485
8	0.693	0.2967	0.5079	0.6066	0.7232
9	0.5896	0.5544	0.5	0.6246	0.7602
10	0.6875	0.5166	0.6212	0.5906	0.5457
11	0.1618	0.5007	0.2875	0.3353	0.1646
12	0.3512	0.2325	0.1104	0.105	0.3839
13	0.1884	0.509	0.1929	0.1027	0.1104
14	0.1869	0.1797	0.6236	0.14	0.3878
15	0.5904	0.0353	0.6234	0.3037	0.1729
16	0.732	0.7485	0.5955	0.6788	0.7345
17	0.5101	0.7571	0.7048	0.7406	0.7753
18	0.6164	0.6416	0.5264	0.7775	0.7035
19	0.6923	0.7988	0.6499	0.629	0.7488
20	0.535	0.6796	0.5664	0.6196	0.5903
21	0.2634	0.2945	0.5018	0.745	0.7073
22	0.2981	0.5209	0.5135	0.274	0.2819
23	0.5783	0.7112	0.2296	0.2082	0.7626
24	0.5232	0.7223	0.2125	0.6331	0.2592
25	0.6909	0.2533	0.5626	0.7021	0.5726

at least $NP = 25,000$. The problem is computationally expensive to solve given very limited computing resources (time and memory). Therefore, the problem requires a modified algorithm to generate a solution.

4.4. **Algorithm: Parallel Differential Evolution.** There are several variations of Parallel Differential Evolution [2] found in the literature and here we have modified and merged the different ones into one suitable for our problem. We implement our version as follows:

- **Step 1:** Request K nodes (or processors) taking one node to be the master node.

TABLE 4. Measures for Similarity $\mathbf{w}_i$ for each attribute by actor: $i = 1, \ldots, 25$

Actor	Education	Age	Income	Politics	Religion
1	0.25	0.5	0.0	0.0	0.0
2	0.25	0.5	0.5	0.0	0.25
3	0.85	0.5	0.25	0.0	0.25
4	0.85	0.5	0.25	0.0	0.0
5	0.85	0.5	0.0	0.5	1.0
6	0.85	0.5	0.25	0.5	1.0
7	0.85	0.5	0.25	0.85	0.25
8	0.5	0.85	0.0	0.0	0.5
9	0.5	0.85	0.0	0.0	0.25
10	0.5	0.5	0.25	0.5	0.25
11	0.5	0.85	0.0	0.5	0.25
12	0.5	0.5	0.5	0.5	1.0
13	0.5	0.85	0.0	0.25	0.0
14	0.5	0.85	0.0	0.85	0.25
15	0.25	0.85	0.0	1.0	0.25
16	0.25	0.85	0.0	0.0	0.25
17	0.25	0.85	0.0	0.0	0.5
18	0.85	0.5	0.5	0.85	0.25
19	0.25	0.85	0.0	0.5	0.5
20	0.25	0.5	0.25	0.5	0.25
21	0.25	0.5	0.5	0.5	0.25
22	0.25	0.85	0.0	0.85	0.25
23	0.25	0.85	0.5	0.0	0.25
24	0.25	0.5	0.25	0.5	0.25
25	0.25	0.85	0.0	0.85	0.25

- **Step 2:** At the master node, create K-1 populations and send one to each of the remaining K-1 nodes.
- **Step 3:** At each of the K-1 nodes, each population evolves toward a nondominated set using DE.
- **Step 4:** As the termination criteria is met, each node sends its nondominated set to the master node.
- **Step 5:** At the master node, compare the K-1 nondominated sets to get the final Pareto-optimal set.

4.5. **Numerical Results and Analysis.** To solve our problem, we used the following criteria:

TABLE 5. Initial rate of change, $\mathbf{v}_i^0$, for each attribute preference by actor: $i = 1, \ldots, 25$

Actor	Education	Age	Income	Politics	Religion
1	0.1216	0.1158	0.1144	0.1116	0.1049
2	0.112	0.1276	0.129	0.129	0.1162
3	0.0394	0.1114	0.1235	0.102	0.1277
4	0.0412	0.1192	0.1123	0.1095	0.1131
5	0.015	0.1052	0.1287	0.1038	0.0054
6	0.0878	0.1236	0.1059	0.1035	0.007
7	0.0022	0.111	0.1159	0.0164	0.1001
8	0.1277	0.0777	0.1151	0.1028	0.1251
9	0.119	0.0342	0.1145	0.116	0.1267
10	0.102	0.1232	0.1113	0.1229	0.1006
11	0.119	0.0148	0.1156	0.1125	0.103
12	0.1226	0.001	0.1245	0.1251	0.0037
13	0.1231	0.0639	0.1063	0.1241	0.129
14	0.1242	0.001	0.1117	0.0009	0.001
15	0.1163	0.0929	0.1014	0.0064	0.1201
16	0.1146	0.0485	0.001	0.1034	0.1137
17	0.001	0.0163	0.1244	0.1065	0.1251
18	0.0626	0.1083	0.1205	0.0909	0.1247
19	0.1023	0.0131	0.1265	0.1043	0.1163
20	0.1163	0.1007	0.1025	0.1062	0.1028
21	0.1063	0.1069	0.1148	0.1255	0.1089
22	0.1081	0.0845	0.1129	0.0713	0.1181
23	0.1088	0.0726	0.1104	0.1024	0.1055
24	0.1247	0.1208	0.1274	0.1246	0.122
25	0.1123	0.0383	0.1048	0.0349	0.1165

- **Requested number of nodes**: 61
- **DE parameters:** $NP = 50$ per node, $W = .5$, and $Cr = .5$
- **Termination Criteria:**
$\sum_{i=1}^{25} \left| \frac{J_i^{(k)}(u^{(1)})+\cdots+J_i^{(k)}(u^{(NP)})}{NP} - \frac{J_i^{(k-1)}(u^{(1)})+\cdots+J_i^{(k-1)}(u^{(NP)})}{NP} \right| < 10^{-5}$

The parameters for DE are based on suggestions from [8] and we created a stopping criteria that drives the problem toward convergence to ensure a good solution.

We use a Pareto optimal trajectory of the MOCP to form our social matrix in Figure 1. By analyzing the relations between actors, we identify two disjoint cliques as shown in Figure 2: **Clique 1** $= \{5, 6, 12\}$ and **Clique 2** $= \{9, 11, 16, 22, 23, 25\}$.

FIGURE 1. Sociomatrix with Interaction Potential, V_{int} ($N = 25$)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1	0	1	1	1	1	1	1
3	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
4	1	0	1	0	1	1	1	0	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	1	0
5	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1	0	1	0	0	1	1	0	1
10	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0	0	0	0	0	1	1	0	0	1	0
11	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	1	0	0	1	0	0	1	1	0	1
12	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	0	1	0	0	1	1	0	1
14	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	1
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1
16	1	1	1	0	0	0	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1
17	0	0	0	0	0	0	0	1	1	0	1	0	1	1	1	1	0	0	1	0	0	1	1	0	1
18	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	1	1	0	1
20	1	0	0	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	0	1	0	0	1	0
21	0	1	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	1	1	0
22	0	0	0	0	0	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0	0	1	0	1
23	0	1	0	0	0	0	0	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	0	1
24	1	0	0	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0
25	0	0	0	0	0	0	1	0	1	0	1	0	0	1	1	1	1	1	1	0	0	1	1	0	0

Figure 3 shows that the social distance between members of Clique 1 is less than the average distance between them. In Figure 4, we see the evolution of the preferences for members in the smaller clique and it is clear that they share similar preferences. Using this preference information along with the model parameters, we can determine the basis for the cliques. For instance, if we look at Clique 1, we conclude that the basis for friendship between actors actors 5, 6, and 12 stems from the fact that their chosen parameters are very repulsive. They have the smallest τ_i, and large weights, $\mathbf{w}_i$ for most categories and their $\mathbf{v}_i^0$ is small for several attribute preferences indicating the lack of motivation to make lots of friends. In fact, they even share similar attribute preferences and demographics. Actors 5 and 6 share four out of five attribute categories; actor 12 shares three out of five attribute categories with actors 5 and 6. All three members of the clique are democrats, college educated ranging in age from $25 - 36$ with religion "other". Now that we know the basis for Clique 1's formation, is it possible to break it? Table 6, suggests that the answer is 'yes'. By definition, a clique requires three mutually friendly actors. From Table 6, we see that if the criteria for friendship becomes slightly stricter, actor 6 no longer perceives that he has two friends and thus the clique is broken. This fact is confirmed in Figure 5 where the social distance between actors 6 and 12 clearly exceeds the average distance between actors. We will explore the possibility of breaking cliques more in Section 6.

5. MEMORY EFFECT

An attractive potential is added to the model that has the same form as the repulsive potential but with opposite sign. With memory effect [3], actors will consider the entire history when deciding to make friends. We expect memory effect to bring

FIGURE 2. Cliques 1 and 2

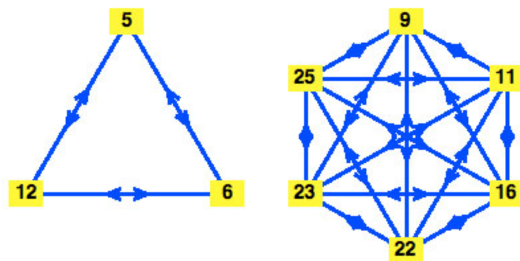
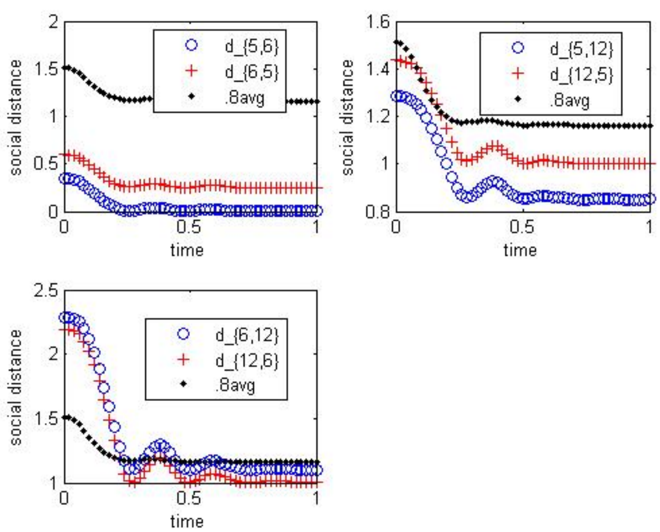


FIGURE 3. Clique 1: Social distance ($d_{ij} \leq .8avg$)



actors who are close together closer making cliques stronger; it also maintains distance between actors who are far apart. Next, we illustrate the impact of the memory potential in the model using a similar multiobjective control approach.

5.1. **Problem Formulation.** The Multiobjective Optimal Control problem becomes: Find the Pareto-optimal set which **minimizes** $\mathbf{J} = [J_1, \ldots, J_N]$ where

$$J_i = \sum_{j \neq i} \|\mathbf{r}_i(t_f) - \mathbf{r}_j(t_f)\|^2 + \sum_{j \neq i} \left\|\mathbf{w}_i(t_f) \cdot (\mathbf{y}_i - \mathbf{y}_j)\right\|^2 + \int_{t_0}^{t_f} \|\mathbf{u}_i(t)\|^2 \, dt$$

FIGURE 4. Clique 1: Evolution of Preferences

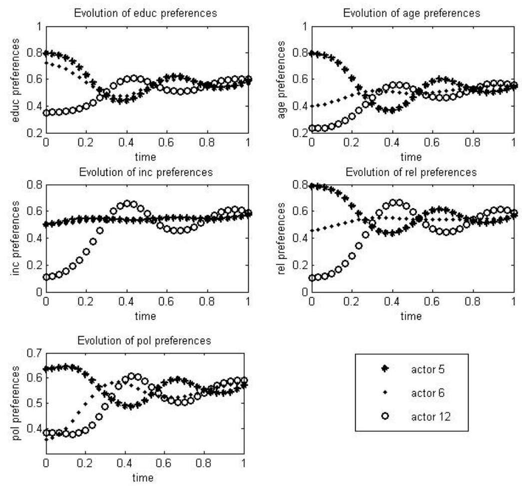


TABLE 6. Number of friends per actor

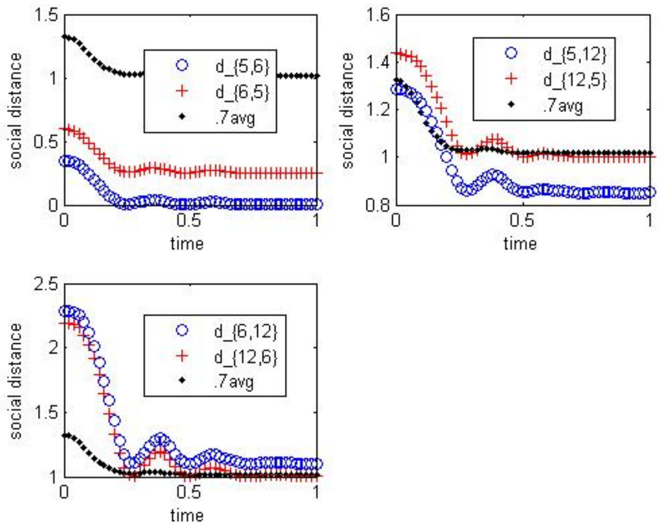
	d_{ij} < avg	d_{ij} < .8avg	d_{ij} < .75avg	d_{ij} < .5avg
1	24	24	24	16
2	19	10	10	2
3	8	5	5	2
4	11	11	8	4
5	2	2	2	1
6	2	2	1	1
7	2	2	2	1
8	8	0	0	0
9	20	13	11	8
10	10	7	7	3
11	15	11	10	2
12	2	2	2	0
13	13	13	10	3
14	10	5	4	2
15	9	4	4	0
16	24	19	14	10
17	16	11	10	6
18	2	1	1	1
19	13	7	7	1
20	18	10	10	4
21	12	8	8	3
22	12	11	5	2
23	16	13	12	4
24	18	10	10	4
25	12	11	5	2

such that

$$\begin{aligned}\dot{\mathbf{r}}_i &= \mathbf{v}_i \\ \dot{\mathbf{v}}_i &= \frac{1}{\tau}(\mathbf{v}_i^0 - \mathbf{v}_i) - \nabla_{\mathbf{r}_i} V_{int} - \nabla_{\mathbf{r}_i} V_m\end{aligned}$$

where V_{int} and V_m are **repulsive** and **attractive** potentials respectively. Simple bounds on state and controls must also be satisfied as before.

FIGURE 5. Clique 1: Social distance ($d_{ij} \leq .7avg$)



The **repulsive potential** for the model has the form:

$$\begin{aligned} V_{int} &= \sum_{j \neq i} \|\mathbf{u}_i - \mathbf{u}_j\|^2 \\ &\cdot (1 + ((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - \mathbf{v}_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2)) \\ &\cdot \exp\{-l_{ij}((\|\mathbf{r}_i - \mathbf{r}_j\| + \|\mathbf{r}_i - \mathbf{r}_j - v_j \Delta t\|)^2 - \|\mathbf{v}_j \Delta t\|^2)\} \end{aligned}$$

The **attractive potential** for the model has the form:

$$V_m(\mathbf{r}_i, t) = \int_0^t \sum_{j \neq i} G(\mathbf{r}, s) \exp\Big\{\frac{t-s}{T}\Big\} ds$$

where

$$\begin{aligned} &G(\mathbf{r}, s) = -\gamma(1 + ((\|\mathbf{r}_i(t) - \mathbf{r}_j(s)\| + \|\mathbf{r}_i(t) - \mathbf{r}_j(s) - \mathbf{v}_j(s)\Delta t\|)^2 - \|\mathbf{v}_j(s)\Delta t\|^2) \\ &\cdot \exp\Big\{ - l_{ij}((\|\mathbf{r}_i(t) - \mathbf{r}_j(s)\| + \|\mathbf{r}_i(t) - \mathbf{r}_j(s) - \mathbf{v}_j(s)\Delta t\|)^2 - \|\mathbf{v}_j(s)\Delta t\|^2)\Big\} \end{aligned}$$

Notice the addition of two more parameters with this new model: γ and T. The parameter, γ, belongs to $[0, 1]$ and reflects how much effect memory has on an actors friendship choice. Large γ indicates that memory has a greater effect when making friends while smaller γ means that the interaction potential plays a greater role in the friendship choice. Memory effect decays at a rate of $1/T$ in the model.

5.2. **Numerical Results and Analysis.** To solve the multiobjective optimal control problem with memory, we choose $\gamma = .2$, $T = .7$ and use the same criteria as before:

- **Requested number of nodes**: 61
- **DE parameters:** $NP = 50$ per node, $W = .5$, and $Cr = .5$

- **Termination Criteria:** $\sum_{i=1}^{25} \left| \frac{J_i^{(k)}(u^{(1)})+\cdots+J_i^{(k)}(u^{(NP)})}{NP} - \frac{J_i^{(k-1)}(u^{(1)})+\cdots+J_i^{(k-1)}(u^{(NP)})}{NP} \right| < 10^{-5}$

To illustrate the effect of adding memory to the model, we examine what happens when actor 25 changes his attribute preferences drastically at some point in time. In Figure 6, we see that **without memory effect**, social distance between actors 25 and 9, 11, 16, 22 fluctuates. Actor 11 is no longer a friend to actor 25 near the end of the time period; in fact, actor 25 is actually out of the clique. Then in Figure 7, we see that **with memory effect**, actors 9, 11, 16, and 22 get even closer to actor 25 after the change. Memory effect enables actors to remember the long-term history of their friends. Clearly, from the graph, **25 is back in the clique** with memory effect in the model.

FIGURE 6. Social Distance for Clique 2 w/o Memory Effect

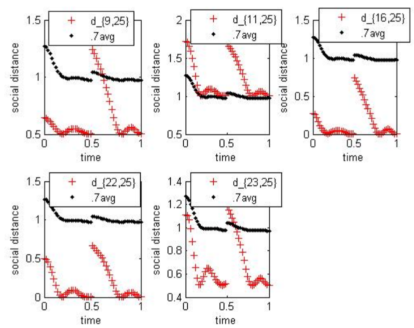
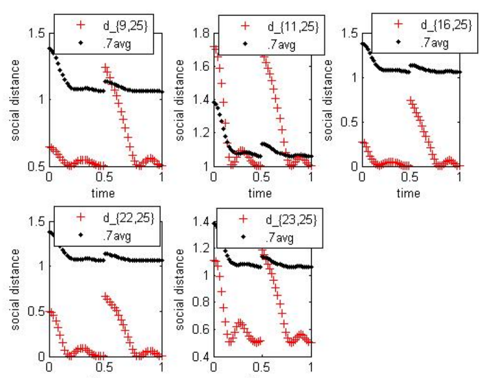


FIGURE 7. Social Distance for Clique 2 with Memory Effect



It is important to note that memory effect can also have the opposite effect of that just discussed. Suppose we assume that at some point in time, actor 12 of Clique 1 changes his attribute preferences drastically. Well without memory effect, the social distance between actors 12, 5, and 6 fluctuates as shown in Figure 8. However, with memory effect, actor 5 remembers that he and actor 12 were friends while actor 6 remembers that they were not friends as indicated in Figure 9. In this case, clearly memory effect brings those who are friends together while keeping those who are not friends apart.

FIGURE 8. Social Distance for Clique 1 w/o Memory Effect

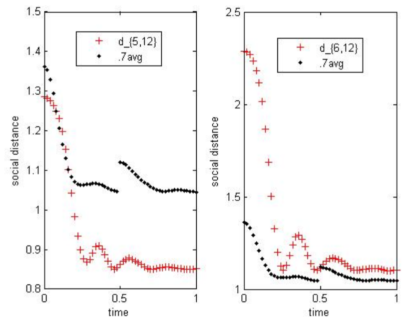
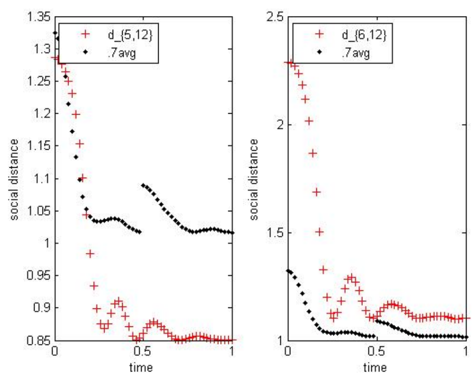


FIGURE 9. Social Distance for Clique 1 with Memory Effect



6. "BREAKING THE TIES THAT BIND"

In this section we explore how our model can address questions of how to bring cliques together and how to break them apart. The answer to such questions could prove crucial to our national security. Suppose we start by addressing the question: is there a way to connect the two cliques? The answer is 'yes'. Since actors 9 and 11 from Clique 2 consider actor 12 from Clique 1 a friend, we should start there. Of course, the fact that actor 12 does not reciprocate their friendship keeps the cliques apart. However, if there was some way to get actor 12 to reciprocate, then the two cliques could connect totally.

We reviewed the model data and parameters used to construct the social matrix over time and determined that changing actor 12's preferences and parameters alone is not enough to connect the cliques. Actor 12 will have to change his preference for diversity, $\mathbf{w}_i$. If he relaxes his weights for categorical similarity on the various attributes, then he will be able to reciprocate the friendship of actors 9 and 11 thus connecting the two cliques (see Figure 10). Figure 11 shows the social distance between these individuals after their preferences/parameters have been altered; clearly, this change enables mutually reciprocated friendship between the actors.

In recent years, the topic of breaking cliques has become even more intriguing than connecting cliques for various reasons. For instance, such a topic appeals to military leaders for its potential to aid in the war on terror. Knowing how to split terror cells would be crucial to our nation's defense. In the example above, we were able to model connecting cliques by changing model parameters and preferences. In the process, we discover valuable information on how to keep the cliques apart. Since the connection of the two cliques centered around actors 9,11, and 12, these actors are indeed potential **"targets"** for disrupting communication between the two cliques. So using our model, not only have we discovered ways to connect the cliques, but we have identified key nodes to focus on when trying to break ties between cliques within a social network. This techniques could prove useful when trying to destabilize a network.

FIGURE 10. Connecting Cliques 1 and 2

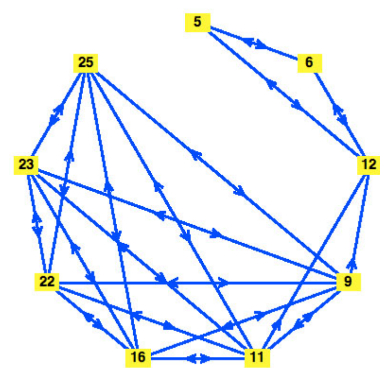
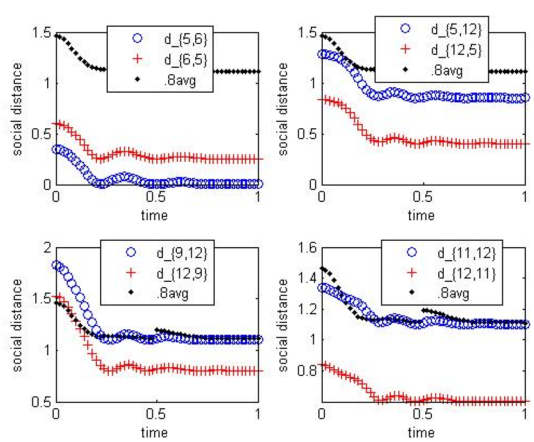


FIGURE 11. Altered social distance between actors 9,11, and 12



7. CONCLUSION AND FUTURE DIRECTIONS

From our work, we conclude that multiobjective optimal control theory provides a suitable framework for the design and prediction of network evolution. Differential Evolution was successfully employed in solving the associated MOCPs with reasonable results. Social force theory with and without memory effect adequately models clique formation as well as suggests potential targets for network destabilization. Since this work focused primarily on cooperative networks like friendships, it will be interesting to research whether or not such models can be refined in the future to include non-cooperative or criminal networks such as terrorist, drug, and other illegal networks. In addition, the model may be able to address such problems as the prediction of missing links. Missing links stem from missing nodes and/or links within a network that for some reason have not yet been discovered. In our upcoming research, we will use known parameters and other existing information from the model to uncover these missing links between nodes.

REFERENCES

[1] Jerry G. Bode, Ronald Burton, John G. Condran, Stephen D. Johnson, David C. Morris, William Morris, Harry Nelson, , and Joseph B. Tamney. Middletown area study, 2004. The Association of Religious Data Archives, www.TheARDA.com, 2004.

[2] V.P. Plagianakos D.K. Tasoulis, N.G. Pavlidis and M.N. Vrahatis. Parallel differential evolution. Technical report, Greece, dtas,npav,vpp,vrahatis@math.upatras.gr.

[3] D. Helbing, P. Molnar, and F. Schweitzer. Computer simulation of pedestrian dynamics and trail formation. May 1998.

[4] Dirk Helbing and Peter Molnar. Social forces model for pedestrian dynamics. *Physics Review*, E(51):4282–4286, 1995.

[5] Johannes Jahn, editor. *Vector Optimization: Theory, Applications, and Extensions*. Springer-Verlag, Berlin, 2004.

[6] George Lietmann, editor. *The Calculus of Variations and Optimal Control: An Introduction*. Plenum Press, New York, 1981.

[7] R. Sarker and H.A. Abbass. Differential evolution for solving multi-objective optimization problems. report, University Of New South Wales, Northcott Drive, Canberra ACT, 2600, Australia.

[8] R. Storn and K.V. Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. *Computer-Mediated Communication*, 11((2006)):1062–1084, 2006.

[9] S. Wasserman and K. Faust, editors. *Social Network Analysis: Methods and Applications*. Cambridge University, New York, NY, 1994.