GENETIC ALGORITHMS IN COLLEGE AND UNIVERSITY HOUSING

Chris Craven
Information Science and Technology Department
Doane College
1014 Boswell Avenue Suite 241
chris.craven@doane.edu

Abstract

Colleges and universities often use a survey process to determine roommate assignments for first-year students. This process is usually done by hand and can take a residence life organization weeks to complete. This project, named Harmony, provides an application which incorporates a custom genetic algorithm to complete this task.

Using these results a residence life staff may be able to see which rooms may experience conflict. Therefore, a residence life office can forecast problems and be prepared in advance of a potential issue. Harmony is available for download on Sourceforge under the GPLv3 license. This project was completed using the RADIS framework which is used by Information Science and Technology students at Doane College to aid in the development of successful projects. The specific phases of the RADIS framework (recognize, analyze, implement and support) will be explained in the presentation of the paper.

Preface

This paper is divided into five major sections marked Recognize, Analyze, Design, Implement, and Support (RADIS). These five sections describe how the project corresponds to the RADIS framework which is used in computer studies at Doane College to aid students in developing successful projects. The sections below describes this project's deliverables for each specific project phase.

Recognize

Many universities and colleges send a rooming survey to incoming students. On this survey the student may indicate what residence hall they prefer or their habits. This information is used to match up potential roommates which will create a stable room environment. This environment will take several major factors into consideration which are believed to create a good foundation for roommates to live together harmoniously. The benefits for the student and potential roommate are clear, however the arrangement also is beneficial for residence life staffs. By reducing the amount of potential conflicts two roommates may have a residence life staff also reduces the amount of conflict resolution and room changes throughout the academic year.

At many colleges and universities this matching of roommates is traditionally done by hand. It often takes an entire residence life staff weeks to find good matches for an incoming student body. There are software programs which can perform this task for the staff as well as interface with the school's student information system, but more often than not these programs are very expensive. This has led to some schools using a custom made solution specific to their school. Neither of these solutions are acceptable for many colleges and universities due to budget constraints or the amount of labor it takes to create custom software.

Analyze

In order to understand how Doane determines if a roommate pairing is compatible Sean Griffin, a Resident Director and Director of Residence Life, was interviewed. Doane considers a roommate pairing compatible when their survey responses are either exactly the same or close to one another. For instance, those who respond as tobacco users should only be placed with others who responded as tobacco users. (Griffin)
In addition to the interview with Mr. Griffin, we researched the resources of the Association of College and University Housing Officers and housing practices at other institutions.

Arizona State University – College of Engineering and Applied Sciences

According to retention studies, the first six weeks for a new student are the most important. Around 50% of the students who end up withdrawing do so during the first six weeks of a new academic term. (Anderson) In the mid to late 1990s Arizona State University began a retention improvement program to increase freshman retention from 68% to 78%. In accordance with this program the College of Engineering and Applied Sciences, CEAS, decided to implement some new measures to increase department retention from 54%. In order to achieve this CEAS specified a dormitory floor for only engineering students. To further increase potential retention a survey was sent to those who opted to live on the engineering floor. The survey, shown in Figure 1, queried the student on items such as study habits, sleep patterns, music preference, etc. Along with the survey came a disclaimer which declared a perfect roommate pairing was impossible but all possible effort would be given to give a good pairing.

FRESHMAN ENGINEERING HOUSING SURVEY - FALL 1998
TO: First Year Engineering and Construction Students
FROM: Office of Student Affairs, College of Engineering and Applied Sciences

You have chosen to live on an engineering dorm floor. In an attempt to make your first year housing at ASU as supportive as possible, please complete the following survey and mail back to the return address given. We will use this information to assist in the assignment of rooms.

Name ____________________ SS# ____________________

Address ____

Phone ____________________ E-mail ____________________

Major ______________ Age______ Class Status________ Gender: ___ F ___ M

In which mathematics class will you be enrolled in the fall?
___ MAT 170 (pre-calculus) ___ MAT 270 (first semester calculus)
___ MAT 271 (second semester calculus) ___ Other:____________

Would you prefer that your roommate has the same major as you do? ______ Yes ______ No
If no, which major(s) would you prefer? ____________________

Do you smoke? _____ Yes _____No
Are you a ______ morning or a ______ night person?
Do you consider yourself athletic? _____ Yes _____ Somewhat _____ No
Relative to study, are you a: ____ hard worker ____ medium worker ____ worker when necessary?
Do you keep your room: ______ tidy ______ somewhat tidy ______ not tidy ?
Do you consider yourself: ______ outgoing ______ flexible ______ reserved?
Do you prefer a roommate who is: ______ outgoing ______ flexible ______ reserved?
Do you require any special accommodations?

What type of music do you enjoy? Check all that apply:
____ easy listening ____ classical ____ country ____ R&B ____ pop
____ classic rock ____ hard rock ____ alternative ____ Other: ____________

List any hobbies or special interests that you have:

Is there anything else about you that might be helpful for us to know in the assignment of a roommate?

Considering the factors that were listed on the other side or any additional factors that you've listed, choose the three factors that would be the most important to you in the assignment of a roommate.

1. ____________________
2. ____________________
3. ____________________

Dormitory Preference (1 for first preference):
____ DORM 1 ____ DORM 2 ____ DORM 3

Figure 1: ASU College of Engineering and Applied Sciences Housing Survey

CEAS used a set of steps in deciding if roommates would be a good fit. First, staff would place the students who had mutually requested each other as roommates together. Second, due to an ASU policy, smokers would be placed with smokers and non-smokers with non-smokers. The third step involved placing two students with the same majors together. If a student couldn't be placed with another student of the same major he or she would be placed in a room that was in close distance of a student with the same major. Third, the rest of the student's attributes on the survey were matched as closely as possible. This was a move to reduce the amount of potential conflict points as pair of students might have. The dormitory of choice was assigned by the date a student sent back the survey.

The results of this initial effort was evident in the increase in retention. CEAS had increased retention from 54% to 82.4%. (Anderson) This was determined from the number of engineering students who had the same roommate from the fall semester during the spring semester. This figure was backed up by remarks made by students in a focus group. These comments included "It helped to have a roommate in the same major as me." and "I knew I had a better chance of getting a good roommate with the survey."

University of Minnesota Study

A study conducted at the University of Minnesota looked into the possible implications a roommate may have on education. Due to the number of roommates the study used only one unit of measure to determine academic ability, a student's high school percentile rank (HSR). The study eliminated all room pairings that had restrictive requests (smoker, specific religious affiliation, etc) and then reduced the number of people in the study to a multiple of eight. The multiple of eight was used in how roommates were assigned. All students were placed in a certain category, categories used are shown in Figure 2.

(1) freshmen in Dorm A
(2) upperclassmen in Dorm A
(3) male freshmen in Dorm B
(4) male upperclassmen in Dorm B
(5) female freshmen in Dorm B
(6) female upperclassmen in Dorm B

Figure 2: Student Categories

Once placed into categories, students were placed in rank order of their HSR. Going down the list every two students were designated as a matched pair. Students above the median HSR were designated as "High" and those below as "Low". From each matched pair students were randomly selected to have a high roommate or a low roommate as a roommate. There was some additional selections made so as to have 24 HH pairs, 48 HL pairs, and 24 LL pairs.

Findings of the study conclude that birth order play a role in roommate relationships. Students who are first-born are more likely to susceptible to be influenced by their roommate. If two first-borns were placed together they benefited from having an overlapping course. In addition, if a low and high student it benefited the pairing if the high-achiever is older than the low-achiever. (Hall 317)

Fazio and Shook Study on Interracial Roommates

One study focused on benefits of interracial roommate relationships. According to a previous study white first year roommates believed they were less compatible with African-American roommates. This belief, while unfounded, has some truth in Ohio State University archival records. 9% of same-race roommates will dissolve compared to 28% of interracial room pairings. All roommates used in the study were randomly assigned. Findings show African-Americans with higher standardized test scores who were assigned a white roommate attained a higher GPA score as shown in Figure 3.

Based on the study, White students receive more of a benefit from the academic abilities of their roommates. An interracial roommate pairing may also help an African-American student acclimate to a predominately white college or university. The major finding of the study is the academic success of an African-American student depends heavily on their sense of belonging. The study did point out that regardless of the higher level of roommate dissolutions among interracial pairings a majority of the pairings were still intact spring semester.

White roommates increased symbolic racism among African-American roommates and created unease among Latino roommates. This was also true for Asian American roommates, whose roommates showed a higher level of symbolic racism. (Shook 430)

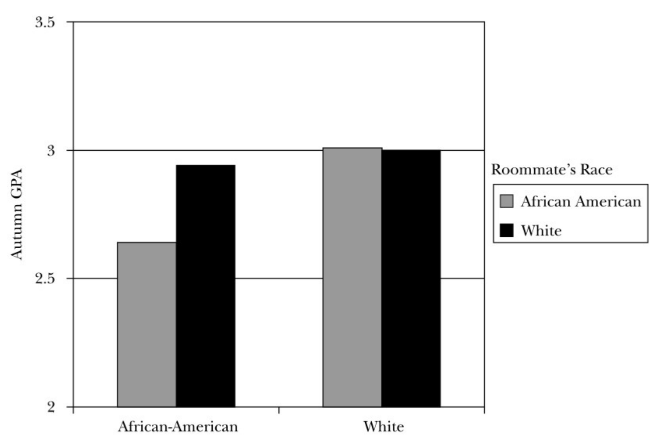

Figure 3: Fall GPA of White and African-American Roommates

Overall Findings

1. The best practice is to not place roommates randomly but to have a designed placing process (Anderson 5)
2. While roommate pairings do have an effect on academics and race relations it is a small effect (Hall 317, Shook 434)
3. Findings from successful rooming assignments from one institution should not be generalized to another institution (McEwan 367)

Design

The type of problem that is posed by placing roommates can be solvable by a genetic algorithm. There are multiple possible solutions but we are looking for a good and acceptable solution. A genetic algorithm evolves a solution over time by simulating a solution as genetic material. This genetic material experiences several simulations of evolution such as mutation, crossover, and selection. The program will maintain a population of x members where each member is a potential solution. A solution, for this project, would be a campus and its attributes. For instance, it would be a campus, its buildings, the levels or floors in a building, the rooms within level of a building, and finally the occupants in those rooms.

There is one technique that has worked well for many colleges and universities, which is matching for compatibility by offering several general questions for the student to answer regarding personal habits to match as closely as possible with another person. This appears to be one technique which has had acceptable success at multiple institutions. The program will still be designed in such a way that a college may redesign the survey to better fit the culture of a campus. The best way for the data to be collected would be through a web form. The data could then be given to the residence life office in comma separated format by the database administrator to be loaded into the program. The survey that will be used is displayed in Table 1.

Question	Data Type Expected
Name	String
Preferred hall	String
Tobacco user	Boolean
Able to live with tobacco user	Boolean
Early riser	Boolean
Night owl	Boolean
Amount of noise can live with (Likert Scale 1-10 1 being low noise, 10 being high noise)	Integer
Room cleanliness (Likert Scale 1-10 1 being not clean, 10 being extremely clean)	Integer

Table 1: Survey and expected data types

Loading in the campus attributes will be similar to the process above. The residence life staff will create a comma separated file with all of the available buildings, levels, and rooms in which a person could be placed. Ideally the residence life office will manually place the exceptions before trying to use the genetic algorithm. These exceptions would include those residents with disabilities, mutual roommate requests, or other potential situations that could not be resolved through the survey.

The heart of a genetic algorithm, the fitness evaluation, will be accomplished using the following method. If all of the roommates are either tobacco users or not tobacco users then the fitness value increases. If one of the roommates is a tobacco user and the other answered that they could live with a tobacco user, the fitness increases. Otherwise the fitness value does not increase. Early riser and Night owl questions are evaluated the same way, either the roommates are all early risers/night owls or they are not in order for the fitness value to increase. Evaluating the two questions with the Likert scale responses assumes that the closer the integer responses are the more compatible the roommates are.

Implement

The program was written in C# using Windows Presentation Foundation(WPF) to aid in UI design. Using WPF can be quickly put together due to a markup code called XAML, which is similar to coding an HTML web page. A solution is represented by classes for a campus, building, level, room. The Campus class contains a list of the Building class, a fitness value of the double data type, and a string name. The Building class contains a list of the Level class, a string name, and a string building identifier. The building identifier is what would be used on the web application that collects survey responses to identify the preferred hall of the respondent. The Level class contains a list of the Room class and an integer identifying what level number it is in a building. The Room class contains a list of survey responses representing residents, a fitness value, an integer room number and integer room capacity, and string gender. The gender attribute determines if the room can hold male or female occupants. Figure 4 shows a UML diagram of the classes described above.

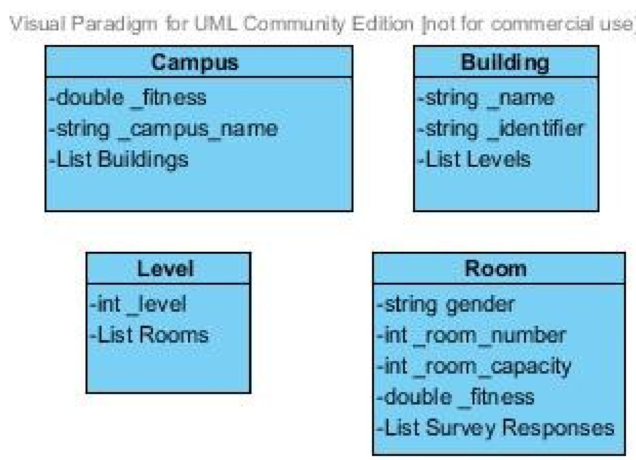

Figure 4: Classes representing a solution or member in the population

When the user starts the program he or she must first create the campus. The initial window that is shown in Figure 5. By selecting File → New Solution a dialog pops up with a textbox for the campus name. Populating the campus with buildings and rooms is also a simple process if you have a properly formatted comma separated file. The button below campus creation in the dialog opens a file selection box. The user chooses the file with their campus buildings and rooms in it and the program will create a campus template for the genetic algorithm to use. Once the campus is completely loaded the user must load survey responses. The process is similar to creating a campus. The user selects File → Add Survey Data which pops up a dialog with a button. Pressing the button opens a file dialog where the user selects the file with the survey responses. When the survey responses are loaded a listbox displays the people that were loaded.

When the steps above are complete, the program is ready to run. The user can adjust the values for x number of population members, the average fitness value required before the run ends, and percent values for mutation and crossover. Pressing the run button will start up the genetic algorithm. On the first run through the algorithm first checks to make sure enough room capacity is available to hold both males and females. If there is not enough capacity the algorithm stops. If the program can continue it generates the first generation for the algorithm to process.

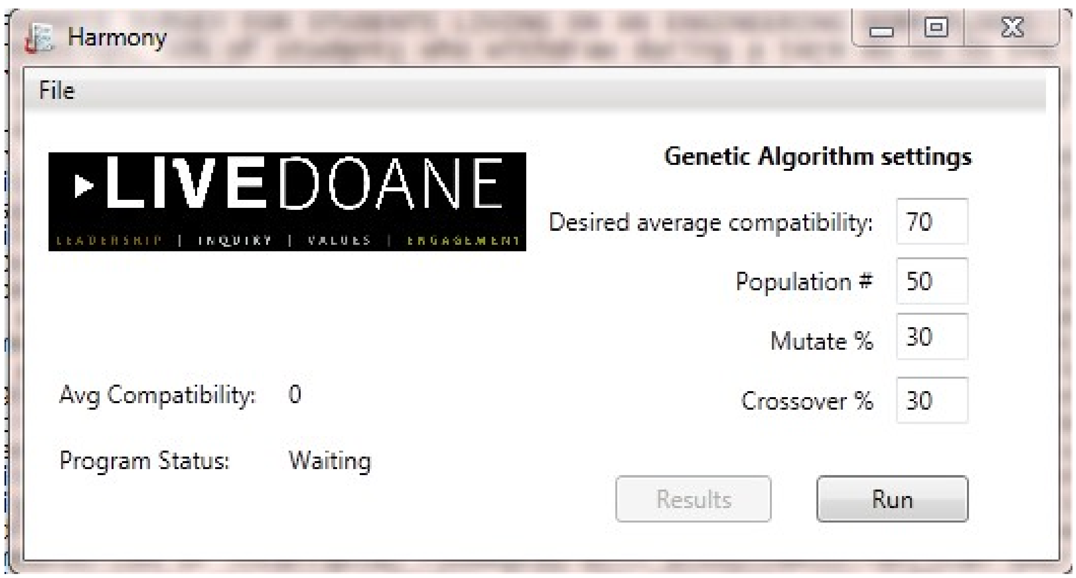

Figure 5: Program after initialization

Creating a member of the population requires creating a new campus by copying the campus template created while loading in the campus file. Once copied, the program iterates through the list of residents. For each resident it first tries to place them in a random room that matches their gender in their preferred hall. If their preferred hall is full it will pick another random hall and another random room. If the random room is full, this process will continue until a suitable room is found.

Once the initial population is created the algorithm will repeat the steps that follow until the required average fitness value is met or the user manually stops processing. The next step that occurs is mutation. Determining if a mutation occurs is done by the use of a random number generator. If the randomly generated number is equal to or less than the percent value specified on the main program window, the mutation occurs. Otherwise, the program will continue without mutating. Assuming a mutation occurs, the algorithm will pick two rooms on the same level of the same building and swap a roommate with the other room. A problem occurs if two rooms of the same gender can not be found on the same level. If that is the case, the mutation method will run until it finds two suitable rooms that are on the same level.

The next step in the program is crossover. Crossover is the simulated exchange of genetic material between members of a population. Determining if a crossover happens is similar to determining if a mutation happens. If crossover occurs, the algorithm exchanges an entire building with two members of a population. A problem that arises with this solution is that a person may be scheduled to be in more than one room at once. To combat this the algorithm will detect if this condition exists and will poison the fitness value of the individual. Poisoning an individual is simply the act of setting their fitness value to a much lower value. This ensures that their genetic material will be passed on but the individual will not be a viable solution.

After crossover, the program evaluates the individuals in the population. In C#, the program uses a method which is called for every room in a solution. In this method the attributes of the potential roommates are checked against one another. The maximum fitness value for an individual is 1.0. Below is how each attribute is checked:

1. To determine if everyone in a room is a tobacco user or not, the program compares the count of the people in the room with the count of those in the room who either use or don't use. This is achieved by calling the list of room occupants and using the count method. To make sure we are only count those who are/are not we use a LINQ method name Count. For example, occupants.Count(`m => m._smoke_chew == true`) will return an integer specifying how many people in the

room are tobacco users. If all of the occupants are either tobacco users or not .3 will be added to the running fitness.

2. If not everyone in the room falls under the two conditions above, the program checks to see if everyone in the room is alright with living with a tobacco user. Using the count method again the program will check to see if everyone in the room is ok living with a tobacco user. If the numbers are equal, .2 is added to the running fitness. Otherwise -.1 is added to the fitness due to tobacco use being a known issue for roommates.
3. Determining if everyone in the room is an early riser is also done by using the count method. If everyone in the room is an early riser .15 is added to the running fitness. Otherwise .05 is added, since waking time is not as important to warrant a negative weight.
4. Determining if everyone in the room is a night owl is completed by using the count method. If everyone in the room is a night owl .15 is added to the running fitness. Otherwise .05 is added, since being a night owl is not as important to warrant a negative weight.
5. Using Doane's method of compatibility, Likert scale responses need to be either the same or close to warrant a successful match. In order to find out how close responses are the program takes the maximum response value from the roommates and subtracts the minimum response value. Using a series of elseif statements, the following is determined. If the difference is less than or equal to two, .2 is added to the running fitness. If the difference is less than or equal to four, .1 is added to the fitness. If the difference is less than or equal to six .05 is added. For a difference of 8, .05 is subtracted and is the difference is less than or equal to 10 .2 is subtracted from the fitness. This method is used on both the noise and cleanliness responses.

After evaluation a simulated selection takes place. The selection is set up tournament style where two random individuals are chosen from the population and the one with the highest fitness value is allowed to continue into the next generation. The loser is removed from the population. This occurs until all members have been tested. After the tournament selection is complete the program generates new members of the population and then repeats these steps beginning at mutation.

Once the program meets the criteria to stop processing the program will be ready to output the best solution. This is done using the PDFSharp library available at http://www.pdfsharp.net under the MIT software license. This library creates a PDF by iterating through the rooms and displaying the room occupants, their survey responses, and the the individual room's fitness. The fitness value for each room will help residence life staffs understand which rooms are at risk for turnover or conflict.

Support

This program, named Harmony, will be released under the GPLv3 license on Sourceforge. There are several features which would make the program more usable for institutions. Firstly, the ability to define how the fitness and survey are defined by using external text files would greatly improve the number of users the program could help. At this time, the fitness and survey can be modified only by changing the source code. These files would be loaded similarly to the way the campus data and survey response data are loaded.

Secondly, there is little support for suites or quads in the program. Theoretically there are several ways that a suite could be defined currently. However, for the average user these approaches will be confusing. Ideally the Room class will be turned into a super class and there could be several room types (suite, double, single) extending the room class. Using polymorphism little of the existing program would have to be modified.

Thirdly, while a desktop application works well in this case there are several benefits to changing it to a web application. The results could be stored in a database which could be updated over several runs during different times. Reporting could be done completely in the browser. Data collection from the residents would be streamlined, allowing a quick turn around time on roommate pairings. The genetic algorithm could run continuously in the background while this data is collected. In addition, using a web browser would necessitate generalizing the parts of the program mentioned in the paragraphs above, making Harmony easier to modify for other institutions to use.

References

Anderson-Rowland, Mary. "Using A Roommate Preference Survey For Students Living On An Engineering Dorm Floor". 1998

Association of College and University Housing Organization. Web. <http://www.acuho-i.org>

Griffin, Sean. Personal interview. 25 Mar. 2011

Hall, L. Robert and Ben Willerman. "The Educational Influence of Dormitory Roommates." Sociometry 26.3 (1963): 294-318

McEwan, J. Patrick and Kristen Soderberg. "Roommate Effects On Grades: Evidence From First-Year Housing Assignments." Research in Higher Education 47.3 (2006) 347-370.

Shook, J. Natalie and Russell H. Fazio. "Roommate Relationships: A Comparison of Interracial and Same-Race Living Situations." Group Processes & Intergroup Relations 11.4 (2008) 425-437.