CS401 FS2004 Exam 2 Key

This is an open-book, open-notes exam. The use of electronic devices is strictly forbidden. Mark all paper you use with your name and the string “cs401fs2004 exam2”. If you are caught cheating, you will receive a zero grade for this exam. The max number of points per question is indicated in square brackets after each question. The sum of the max points for all the questions is 140, but note that the max exam score is capped at 100 (note that answering three of the four questions completely correctly is more than enough for a perfect score of 100!). You have exactly 50 minutes to complete this exam. Good luck!

1. You are hired by the severely understaffed admissions office of a major university to design an automated screening process which receives electronic datafiles on applicants as input and gives admission advice as output. Typical types of advice are: reject, accept, investigate further, etc. You have access to all the experienced admissions office staff as well as many previous years of applicant files which contain applications as well as admission decisions. Outline in detail your EA design to tackle this problem. [35]

This is a type of modeling problem (page 9 of textbook). You have historical input data (application files) and output data (admission decisions) which describe the behavior of an expert system (the admissions office staff). What you want to obtain is a system which models this behavior in order to accurately predict future admission decisions. Such a system can be used to process applications, leaving only undecidable ones for the human experts to investigate. A type of EA well suited for this problem is a fuzzy LCS. An LCS would lend itself to this type of problem because you are basically dealing with a rule-based classification problem with a very limited number of possible actions (reject, accept, undecided). A fuzzy LCS is indicated because the condition part of some of the rules will be fuzzy (e.g., large amount of extracurricular activities AND low amount of charitable work). With this problem it is very likely that rule sets will evolve which often result in conflicting actions, therefore a well-thought out conflict resolution mechanism will be of paramount importance; one possibility is majority vote. Finally, for the initialization phase you can utilize the expert knowledge of the staff by capturing it in the form of rules with which to seed the initial population.

2. You are hired by the overworked chair of a Computer Science department to design an automated course scheduling program which receives as input the list of sections to be taught, the list of instructors, a can-teach relation between the instructors and the sections, a want-to-teach relation between the instructors and the sections, a list of time slots, a can-be-available relation between the instructors and the time slots, and a preferred-time slot relation between the instructors and the time slots. You do not have to worry about how you schedule sections relative to each other nor where they will be taught. Outline in detail your EA design to tackle this problem, enforcing all hard constraints while keeping the instructors as happy as possible. [35]

This is a classic schedule optimization problem (see for instance Section 3.9 in your textbook). One possibility for genotype is a sequence of lists, with one list per time slot and each list consisting of [section, instructor] pairs. The hard constraints in this problem are the can-teach and can-be-available relations, as well as assigning an instructor to each section, not assigning the same instructor to more than one section per time slot, and not assigning the same section multiple times in the same time slot. While the genetic operators could be made such that they never produce individuals which violate any of these hard constraints, the complexity of the problem space indicates the use of penalty functions instead (utilizing for instance the second option for changing penalty coefficients as detailed on page 135 of the textbook). The combination of the soft constraints (the want-to-teach and preferred-time slot relations) is an obvious candidate for fitness function: the more soft constraints fulfilled, the higher the fitness. Finally, the schedules from preceding years can be used to seed the initial population.
3. You have decided you would prefer to skip the decades of hard work between college and retirement by applying your vast knowledge of EC to making a kill on eBay. Your idea is to create an autonomous program which will exclusively buy and sell items on eBay to generate profit. Outline in detail your EA design of this program. [35]

This is a type of modeling problem (page 9 of textbook). Given a certain amount of start capital (in the form of cash), pricing information on a variety of goods (from Amazon.com, Pricewatch.com, etc.), and sensors to determine the state of eBay at any given time (i.e., data on what is for sale, bidding histories, etc.), make bidding decisions (i.e., for all items for sale decide whether to bid for them and if so how much). The termination condition is running completely out of money as well as goods to sell, otherwise the cycle continues indefinitely (well, as long as eBay exists). One time step works as follows: if you own goods, then you need to decide whether to offer them for sale and if so for how much; if you have cash, then you need to decide whether to bid for items on sale and if so how much. The most obvious type of EA to use here is GP, as it is well suited to taking various variables as input, performing a calculation based on those variables, and outputting the result (a bid or sell amount, with zero indicating no bid nor sale). The fitness function can naturally be defined by the amount of money in the program’s bank account (perhaps in combination with a conservative estimate of the value of the program’s inventory).

4. You have seen the light and joined UMR’s ACM SIG Security. It is time to go Wardriving and you need to build a cantenna. Any can (e.g., can of vegetables) will do, but for best reception you need to optimize some complex set of equations describing the center point of the antenna. There are two parameters that you can easily influence by your selection of the can, namely can diameter and height; however, both diameter and height are constrained by two sets of inequalities based on can availability and certain physical properties of antennas. Outline in detail how you would use an EA to optimize the can diameter and height under these constraints so that you know which can to buy at your favorite grocery store. [35]

This is a classic real-valued constraint satisfaction optimization problem of the type that traditionally Evolution Strategies were used for. An individual’s genome would encode the diameter and height parameters with two floats. Standard Gaussian mutation and uniform crossover should work just fine for this problem. While genetic operators could be employed which are guaranteed to only produce legal solutions, it may be necessary based on the quality of the results to use a penalty function instead (utilizing for instance the second option for changing penalty coefficients as detailed on page 135 of the textbook). The boundary constraints such as not requesting cans with negative diameters and/or heights should probably be enforced at all times. Finally, to significantly speed up finding good solutions it may be a good idea to seed the initial population with the dimensions of actual cans sold at your favorite grocery store.