William Martin
Homework 6
CSC761, Advanced Artificial Intelligence
Dr. Manual Penaloza
Spring Semester, 2003

Grade: _____

Problem Description

Do a research on the following issues and questions and write your findings. Include references.

Questions:

a) Is it beneficial to maintain a constant population size?
b) What is overfitting? How can it be avoided?
c) Explain a problem (other than those in the class textbook and TSP) that has been solved using GA. Describe as much as possible, the problem, individual coding, the fitness function, the GA operators, and results.

Solutions

a) Is it beneficial to maintain a constant population size? Yes.

Why?

- It is beneficial because of "sampling disruption": as the size decreases (i.e. smaller population), it will be too small to provide the necessary sampling accuracy for complex search spaces. [4] It lacks the information capacity to provide accurate sampling.

- As the size increases (gets larger), we dilute our sampling space with a few different ways:

  (1) If we introduce only strong members, then the population is biased.
  (2) If we introduce only weaker members, then the population is biased.
  (3) If we add random members (weak, strong, medium), we run the risk of diluting the evolution process we've already started.

  In all cases, when using probabilities for crossover and mutation, a strong member becomes less likely to be chosen without consideration towards the

- As the size increases, it will take longer to converge to global optima.

However, it is useful to note that smaller population sizes (i.e. remain constantly small), the sampling space becomes homogeneous faster.

A convergence is determined for a particular gene when 95% or more of the population have the same value (i.e. same chromosome). A global convergence occurs when all of the chromosomes have converged. [6]

b) What is overfitting? How can it be avoided?

Overfitting is the process of focusing on the training examples so much that your solution is no longer accurate for all real instances.

In Genetic Algorithms, overfitting is evolving a population to a point where the correct solution can no longer be "discovered". We can overshoot the global optima. This occurs mainly because our solution space is no longer generalized.

It can be avoided by a few methods:

1. **Stop evolving early.** By changing the stopping criterion, we may be able to reduce or eliminate overfitting. [9]

2. **Penalizing classifier complexity**, which would penalize overly large networks.

3. **V-Fold Cross Validation**. [9]: From [2], simple validation is the process of randomly splitting the set of labeled training samples D into two parts: one is used as the traditional training set for adjusting model parameters in the classifier. The other set – the validation set- is used to estimate the generalization error. Essentially, the Genetic Algorithm will have a set that is used for the population, and the validation set. Once this error has received a specified lowest threshold, we can stop evolving, i.e. report the solution, but prevent overfitting.

So, it is very important that the criteria for stopping be scrutinized beforehand. This can be done by empirical testing results.

P.377, Reference 2

C) Article Discussion

Explain a problem (other than those in the class textbook and TSP) that has been solved using GA. Describe as much as possible, the problem, individual coding, the fitness function, the GA operators, and results.

Color Image Segmentation with Genetic Algorithm for In-Field Weed Sensing [5]

Link:http://age-web.age.uiuc.edu/remote-sensing/Papers/gahsi-final-with-figures.pdf

Document conventions

In my solution to question **c**, I'm discussing a little bit about the paper, but more importantly outlining the aspects of this paper that relate to Genetic Algorithms. To make reading easier for the reader, I've **bolded** the items that discuss aspects of Gas.

Introduction

The choice to use this paper was based on the fact that it produced the most information regarding how it implemented the GA, covering most or all of the concepts learned in class. I also was interested in Remote Sensing applications, so I investigated this idea by searching on Google for "Remote Sensing Genetic Algorithms".

The objective of the paper's study was to test the possibility of using a GA to detect a relatively stable color region in H.S.I (Hue Saturation and Intensity) color space to segment vegetation under two extreme outdoor field lighting conditions: cloudy and sunny sky conditions effect on images.

Image Segmentation refers to subdividing an image into parts, which are organized according to similar characteristics. Algorithms performed on these parts include feature extraction and object recognition, which depend entirely upon the "goodness" of each of the subdivisions.

Coding

The search space is defined as a matrix of six variables, which are the upper and lower bounds of the hue, saturation, and intensity. Each boundary pair has 256x256 combinations, which equates to an possible total of 2.8 x 10^14 boundary combinations. An efficient searching algorithm is needed to solve this problem.

The **population** size was determined using D.E. Goldberg's [6] rule of thumb, which stated that the size of the population should be equal to the **chromosome** length (the "string" of bits, for example). With this in mind, the authors of this paper decided that the size should be 48. They chose 48 bits for a chromosome size, using these bits to represent the "plant" region boundaries in H.S.I space. Each boundary parameter is one byte-long and is assigned a location in the chromosome bit string. The locations are very important because the genetic algorithms have a special method of choosing combinations of parameters [6]. The makeup of the entire chromosome string is as follows:

Byte 1: The upper hue boundary.
Byte 2: The lower hue boundary.
Byte 3: The upper saturation boundary.
Byte 4: The low saturation boundary.

Byte 5: The upper intensity boundary.
Byte 6: The lower intensity boundary.

The three pairs of boundaries are in adjacent positions because they belong in the logical position of the H.S.I model.

Selection

The Selection process in genetic algorithms is based on the fitness value of an individual in relation to the rest of the population. This paper uses a local tournament selection algorithm, which is, selecting individuals from the current generation. This selection, as discussed in our textbook [7], is based on a higher probability, p, of the individual(s) with the highest fitness values, and the inverse, 1-p, of the individuals with the lower fitness values.

They chose the tournament selection for the reason that the roulette wheel selection can cause premature problems at earlier stages. They also use a no replacement scheme, with a size of 4.

Crossover

Crossover is the process of producing two new children chromosomes from two parent chromosomes. The location of where the bits are taken from in the parents is a predetermined index. De Jong [8] showed that good GA performance requires a high crossover probability. This paper uses a value of 0.8.

Mutation

Mutation is the process of selecting one or more bits for inversion in the children after crossover has occurred. The outcome of performing mutation provides for occasional disturbances; a method that attempts to mimic natural evolution. Goldberg recommends that the mutation rate is inversely proportional to crossover.

The authors have noted that convergence occurs more rapidly using tournament as their selection choice, so instead of using a value of 0.02, they have increased it to 0.03. It is unknown whether there was a mistype as the crossover rate was determined to be 0.8, and I made the assumption that the crossover and mutation should equal to 1.

Fitness Value

The fitness of an individual is determined by the performance of its phenotype. That is, how well does the current chromosome (bit string) match the solution vector. This article uses a reference image to "gauge" the fitness of the generated segmented pixels.

Segmentation was used as a performance measure for function evaluation. They used two performance measures, object sensitivity (SenO) and background sensitivity (SenB).

SenO represents the ratio of correctly segmented plant pixels in the test image to the total number of plant pixels in the reference image.
SenB represents the ratio of the number of correctly segmented background pixels in the test image to the total number of background pixels in the reference image.

$$SenO = \frac{C_p}{C_p + I_B} \qquad SenB = \frac{C_B}{C_B + I_P}$$

Where:

$C_p = P - I_B$ =Number of correctly classified plants in the test image with respect to the reference image.
P = total number of plant pixels in the reference image.
I_B = Number of pixels classified as plants in reference image, but as background in the test image.

I_P = number of pixels classified as background in reference image, but as plants in the test image.
$C_B = B - I_P$ = Number of pixels correctly classified as background in the test image with respect to the reference image.
B = Total number of background pixels in the reference image.

When the researchers used the average of SenO and SenB as the fitness evaluation, they noticed it was difficult to reduce the amount of noise in the background after segmentation. Thus, given the fact that the background pixels and objects are very different from image to image, they used a weighted average (biased fitness) of SenO and SenB, which was based on their corresponding pixel ratio in the image.

The equally weighted fitness equation is: **FE**= 0.5 * SenB + 0.5*SenO
The biased fitness is: **FB**= Oratio * SenB + (1-Oratio) * SenO.

Where Oratio is the ratio of the object pixel number to the total pixel number in the reference image. The Oratio was determined to be 0.2384 in a mosaic image. A mosaicked image was a combination of four images selected from a sampling of images, with two being from completely sunny images and two under cloudy conditions.

Stopping Criteria

A genetic algorithm is said to have converged when it has reached a global optima. The question is, how does one derive this value? Many algorithms have a base rule of stopping after a certain number of iterations have occurred. How this number is determined can be done through empirical testing.

This paper discusses three ways in which it decides that it has exhausted itself to the point of finding global optima. They are:

(1) If a "Utopian parameter set" is located such that the fitness value has reached a point above a predefined threshold. The paper uses the value of 99 % with respect to hand-segment generated reference images.
(2) If the fitness value hasn't improved over 5 consecutive generations.
(3) If the iteration, or generation, number has reached 100.

If any of these conditions are met, the best-fit chromosome string is decoded as the boundaries in H.S.I color space of the "plant" region.

Outcome and Findings

The authors experimented with many factors, such as changing the size of population. They noticed there was no possible gain from increasing the size of the population to anything above 100. The also change the probability of crossover, to the extremes of 0 and 1 with no observed final improvement. When they removed crossover from the entire algorithm, they were able to obtain the best results, however the convergence rate was noticeably slower.

They also went on to discuss the boundaries of the H.S.I model and the effect they had on the segmentation performance. The lower boundaries of the color model seemed to have the most significant effect.

Conclusion

In conclusion, the paper reports that they have proven that the GA method described in their paper was a viable method for machine vision-based weed sensing in variable lighting conditions. There were several factors, such as conversion from RGB to H.S.I model, that were explained as having direct effects on the recognition of segmented images and cracks in the soil. These cracks were discovered to be plant regions and, thus, misclassified.

They recommend further testing with different imaging devices and color transformations, and, more importantly for my findings, GA codings and operators.

References

[1] Penaloza, Manuel, Dr., "*Advanced Artificial Intelligence*", South Dakota School of Mines and Technology, Spring 2003

[2] Duda Richard, Hart E. Peter, David G. Stork, "*Pattern Classification*", © 2001, John Wiley & Sons, Inc

[3] Damian Eads, Daniel Hill, Sean Davis, Simon Perkins, Junshui Ma, Reid Porter, and James Theiler, "*Genetic Algorithms and Support Vector Machines for Time Series Classification*", http://www.cs.rit.edu/~dre9227/papers/eadsSPIE4787.pdf, Los Alamos National Laboratory and Rochester Institute of Technology

[4] http://www.aic.nrl.navy.mil/~spears/papers/ppsn90.pdf

[5] Tang, Tian, Steward, "*Color Image Segmentation with Genetic Algorithm for In-Field Weed Sensing*", University of Iowa, http://age-web.age.uiuc.edu/remote-sensing/Papers/gahsi-final-with-figures.pdf

[6] Goldberg, D.E. 1989a. "*Genetic algorithms in search, optimization and machine learning*.", Mass.: Addison-Wesley Publishing Company, Inc.

[7] Mitchell, Tom, "Machine Learning" Textbook, McGraw-Hill, © 1997

[8] De Jong, K., "*An Analysis of the Behavior of a Class of Genetic Adaptive Systems*", PhD Dissertation, Department of Computer and Communication Sciences, University of Michigan, Ann Arbor.

[9] Chen, Shu-Heng, Chen, Chien-Fu, Ching-Wei, Tan, "*Toward an Effective Implementation of Genetic Algorithms in Financial Data Mining: Retraining plus Validating*", National Chengchi University, Taipei, Taiwan, 1998, http://aiecon.org/staff/shc/vita%5CIDEAL983.pdf

Other (Interesting) Potential Articles to Discuss in Question C

CIRCUIT SYNTHESIS EVOLUTION USING A HARDWARE-BASED GENETIC ALGORITHM
Link: http://www.site.uottawa.ca/~rabielmo/gas/227.pdf

Protein Folding Simulation With Genetic Algorithm and Supersecondary Structure Constraints
http://www.biostat.harvard.edu/complab/Protein%20Folding%20Simulation.pdf

Physically Embedded Genetic Algorithm Learning in Multi-Robot Scenarios: The PEGA algorithm
http://cswww.essex.ac.uk/staff/udfn/ftp/epirob03.pdf

Image Processing algorithm for matching horizons across faults in seismic Data
http://isgwww.cs.uni-magdeburg.de/bv/pub/pdf/IAMG_Melanie.pdf