Collected And Created By AMI MASUD

Uniform Crossover

If the mixing ration is 0.5 approximately. then half of the genes in the off spring will come from parent 1 and other will come from parent 2.

P1 : 1101100100110110

P2 : $\boxed{1101.11.1\ 000.0\ 111.10.}$

OS1 : $1_1 \boxed{1_2 0_2} 1_1 1_1 \boxed{1_2 1_2 0_2} 0_1\ 0_1 \boxed{0_2} 1_1 \boxed{1_2} 1_1 1_1 \boxed{0_2}$

OS2 : $\boxed{1_2} 1_1 0_1 \boxed{1_2\ 1_2} 0_1 0_1 1_1 \boxed{0_2\ 0_2} 1_1 \boxed{1_2} 0_1 \boxed{1_2\ 1_2} 0_1$

Arithmatic Crossover

$OS1 = a * Parent1 + (1-a) * parent2$

$OS2 = (1-a) * parent1 + a * parent2$

where a is a random weighting factor chosen before each crossover.

P1 : (0.3) (1.4) (0.2) (7.4)

P2 : (0.5) (4.5) (0.1) (5.6)

here, $a = 0.7$

OS1: $0.7 * 0.3 + (1-0.7) * 0.5 = 0.36$

OS2: $(1-0.7) * 0.3 + 0.7 * 0.5 = 0.44$

OS1: $0.7 * 1.4 + (1-0.7) * 4.5 = 2.33$

OS2: $(1-0.7) * 1.4 + 0.7 * 4.5 = 3.57$

: : - - - - - - - - - -

Final:

OS1 : (0.36) (2.33) . . .

OS2 : (0.44) (3.57) . . .

Heuristic crossover

OS1 : Best Parent + r * (Best Parent - Worst Parent)

OS2 : Best Parent.

here r is a random number between 0 and 1

Mutation:

alters one or more gene values in a chromosome.

Heuristic crossover

OS1 : Best Parent + r * (Best parent - Worst Parent)

OS2 : Best Parent.

here r is a random number between 0 and 1

Mutation :

alters one or more gene values in a chromosome.

Adversarial Search

Double or multi agent.

$\boxed{\text{Deterministic, fully observable}}$

(*) Componant of games:

Initial state

Player(s)

Action(s)

Result(s)

Terminal-test(s)

Utility functions

Proparties of minimax

$O(b^m)$ b = branching factor
m = depth of the treee

Heuristic evaluation f^n

$E(n) = M(n) - O(n)$

(*) α, β Pruning:

Alpha (α) → the value of the best choice so far along the path for (MAX).

Beta (β) → the value of the best choice (lowest) so far along the path for (MI

(*) first min তার থেকে বড় কোন value নিবে না

(*) first max " " ছোট " " নিবে "

for α = - infinite

β = + infinite

prune whenever $\alpha \geq \beta$

CSP (Constraint Satisfaction Problem)

Componants of CSP: (Domain, Variables, Constraint)

Varieties of CSP:

- Discrete variables
  - Finite domains
  - Infinite "
- Continuous variables

Varieties of Constraints:

Preferences (soft constraints)

rules -মেট meet হলে ভালো, কিন্তু meet হচ্ছে না কোন কারণে we can skip.

Unary (single variable) SA ≠ green

Binary (Double variable) SA ≠ WA

Higher-order (three or more)

⊛ Backtracking Search:

#Idea1: only consider a single variable at each point.

#Idea2: only allow legal assignments at each point.

⊛ Improvement on backtracking search

1. MRV (minimum remaining value) - variable ordering - fail first.

2. Degree heuristic - variable ordering

3. Least constrained value / Least remaining constraine.

Statistical Reasoning

Classical Probability = a priori theory of Probability

P(A) = f / n

where f = total number of possible outcomes

and n = number of outcomes

Conditional Probability = Some event A, given the occurrence of

some other event B.

P(A|B) read as "The probability of A, given B"

Joint Probability = both events together

P(A,B)

Marginal Probility = P(A), P(B)

Disjoint Events = Mutually Exclusive Events means nothing in

common. They can not occur at the same time.

P(A and B) = 0

Either occuring is P(A or B) = P(A) + P(B)

The non mutually exclusive events have some overlap.

P(A and B) = some value

All Probabilities are between 0 and 1 inclusive 0 <= P(E) <= 1

The probability of any event which is not in sample space is 0

The probability of an event not occuring is $P(E') = 1 - P(E)$

If A and B are independent = $P(A \text{ and } B) = P(A) * P(B)$

If A and B are not independent = $P(A \text{ and } B) = P(A) * P(B|A)$

where $P(B|A)$ is conditional probability of B given A

A standard deck of playing cards consists of 52 cards.

All cards are divided into 4 suits.

There are two black suits — spades (♠) and clubs (♣) and

two red suits — hearts (♥) and diamonds (♦).

In each suit there are 13 cards

including a 2, 3, 4, 5, 6, 7, 8, 9, 10, a jack, a queen, a king and an ace.

If A and B are mutually exclusive = $P(A \text{ or } B) = P(A) + P(B)$

If A and B are not mutually exclusive

= $P(A \text{ or } B) = P(A) * P(B|A) - P(A \text{ and } B)$

Where $P(A \text{ and } B)$ are independent

and $P(B|A)$ is conditional probability of B given A

Product Rule:

1) $P(A \wedge B) = P(A) * (B|A)$

2) $P(A \wedge B) = P(B) * (A|B)$

Bayes Rule: $P(B|A) = (P(B) * P(A|B)) / P(A)$

$P(\text{cause}|\text{effect}) = (P(\text{cause}) * P(\text{effect}|\text{cause})) / P(\text{effect})$

P(effect|cause) = casual direction

P(cause|effect) = diagnostic direction

Bayes theorem is used to calculate conditional probability.

Bayesian Networks could represent the probabilistic

replationship between diseases and symtopms.

Bayesian Networks are also called Influence Diagrams.

Name of Bayasian network: 35 page

Bayasian network consists of: page 36

Agents may need to handle uncertainty, whether due to partial observability,

nondeterminism, or a

combination of the two.

Decision theory = probability theory + utility theory

Rational = maximum expected utility (MEU).

CSP

CSPs are specilized for identification problems

CSP is simple example of a formal representation language

Latin Square, Eight queen, Sudoku are CSP problem

Latin Square: In n*n grid, each symbol occurs exactly once

in each row and each symbol.

N queen: In n*n grid no queen attack each other in row,

column, and diagonal.

Variable: states, Domain: colors/numbers

Discrete Variables:

Finite Domain: size is O(d^n) and NP-Complete

Infinite Domain: Job scheduling, linear contarint solvable

Continious Variables:

Hubble Telescope observations

Linear containts solvable in polynomial time by LP methods

Simplest CSP ever: two bits, constrained to be equal.

DFS for CSPs with below two improvements is called Backtracking:

1. Only consider a single varibale at each point

2. Only allow legal assignments at each point

Can solve n-queen for n = 25

Backtracking = DFS + variable odering + fail on validation

Forwad checking propagates information from assigned to unassigned variables, but does not provide early detection for all failures.

ARC - an arc C -> Y is consistance iff for everu x in the tail there is some y in the head which could be assigned without violating a constraint. O(n^2*d^2)

MRV - chooses the variable with the fewest legal values. It also has been called the “most constrained variable” or “fail-first” heuristic

Orderning - Least constraining value. Cobining these heuristics makes 1000 queens feasible.

In CSP, States defined by values of a fixed set of variables. Goal test defined by constraints on a variable values.

CSP general-purpose rather than problem specific heuristics

Backtracking algorithms are the basic uninformed algorithms for CSPs

Can solve nqueen for 25

Adverisial Search

Mathematical game theory, a branch of economics, views any multiagent environment as a game.

Deterministic and full obeservable environments in which two agents act alternately and in which the utility values at the end of the game are always equal and opposite.

Chess as a First Choice

Some games can normally be defined in the form of a tree.

Branching factor is usually an average of the possible number of moves at each node.

##Initial state: Set-up specified by the rules,

e.g., initial board configuration of chess.

##Player(s): Defines which player has the move in a state.

##Actions(s): Returns the set of legal moves in a state.

##Result(s , a): Transition model defines the result of a move.

##Terminal-Test(s): Is the game finished?True if finished, false otherwise.

##Utility function(s , p): Gives numerical value of terminal state s for player p.

E.g., win (+1), lose (-1), and draw (0) in tic-tac-toe.

E.g., win (+1), lose (0), and draw (1/2) in chess.

Two players: Max and Min

Objective of both Max and Min to optimize winnings

Max must reach a terminal state with the highest utility

Min must reach a terminal state with the lowest utility

Game ends when either Max and Min have reached a terminal

state

Max must develop a strategy that determines best possible

move for each move Min makes.

Minimax Algorithm determines optimum strategy for Max.

Complete?

Yes (if tree is finite).

Optimal?

Yes (against an optimal opponent).

Can it be beaten by an opponent playing sub-optimally?

No. (Why not?)

Time complexity?

O(bm)

Space complexity?

O(bm)   (depth-first search, generate all actions at once)

O(m)   (backtracking search, generate actions one at a time)

Cutoffs must be implemented due to time restrictions,

either by computer or game situations.

The performance of a game-playing program is dependant

on the quality of the evaluation functions.

Heuristic is E(n) = M(n) - O(n)

M(n) = is the total of My possible winning lines

O(n) = is the total of Opponent's possible winning lines

E(n) = is the total Evaluating for state n

The deeper the search the more information is available

to the program the more accurate the evaluation functions

Evaluation function might return an incorrect value.

If the search in cutoff and the next move results involves

a capture, then the value that is return maybe incorrect.

Improvements to Cutoff = Quiescence search.

The process of eliminating a branch of the search tree

from consideration without examining it.

Alpha Beta Pruning = Returns the same choice as minimax

cutoff decisions but examines fewer nodes.

Alpha –

the value of the best choice so far along the path for MAX.

Beta –

the value of the best choice (lowest value)

so far along the path for MIN.

Prune whenever alpha >= beta.

Max nodes update alpha based on children's returned values.

Min nodes update beta based on children's returned values.

Repeated states are again possible.

Store them in memory = transposition table

If there is only one legal move, this algorithm will

still generate an entire search tree.

Genetic Algorithm

Local search algorithms operate using a single current node (rather than

multiple paths) and generally move only to neighbors of that node

There are many local search algorithms namely,

Hill-climbing Search, Simulated Annealing, Local

Beam Search, and Genetic Algorithms.

Binary encoding gives many possible chromosomes even

with a small number of alleles ie possible

settings for a trait (features).

Genetic algorithm are Adaptive heurestic search algorithm

based on the ideas of natural selection and genetics.

Genetic algorithm are part of Evolutionary computing inspired

by Darwin's theory of evolution - "Servival of the fittest".

Fittest individuals dominating over the weaker ones.

GAs are the ways of solving problems by mimicking processes

nature uses., Selection, Cross over, Mutation and Accepting.

GAs are intelligent exploition of random search used in

optimization problems.

Organism -> cells -> chromosomes -> genes -> trait -> alleles -> locus

A Gene represents some data (eye color, hair color, sight)

A chromosome in an array of geens.

In binary encoding every chromosome is a string of bits : 0 or 1

Permuatation encoding can be used in ordering problems, such

as travelling salesman problem or task ordering problem.

Tree encoding is good for evolving programs. The programming

language LISP is often used.

REPRODUCTION (SELECTION)

CROSSOVER (RECOMBINATION)

MUTATION

**--> Roulette Wheel Selection

Pi = Fi / sum of F1 to Fn

Porbability = Single fitness / Total fitness

Uniform Crossover: If the mixing ratio is 0.5 approxinately,

then half of the genes in the offspring will come from parent

1 and other half will come from parent 2.

Arithmetic Crossover:

Offspring1 = a * Parent1 + (1-a) * Parent2

Offspring2 = (1-a) * Parent1 + a * Parent2

where, a is a random weighting factor

Flip Bit Mutation: The mutation operator simply inverts

the value of the chosen gene. 0 goes to 1 and 1 goes to 0.