Learning classifier systems

- Developed: USA in the 1980's
- Early names: J. Holland, K. de Jong
- Typically applied to:
  – machine learning tasks (prediction, classification…)
- Attributed features:
  – represents & generates knowledge in rule format
  – "just a branch of GAs"
- Special:
  – GA is only a part of the whole, the discovery system

Evolutionary Computing   Learning classifier systems   1

Learning classifier systems

Classifier systems main propeerties:

- sub-branch of genetic algorithms
- to develop rule sets for performing a task
- basic rule format:

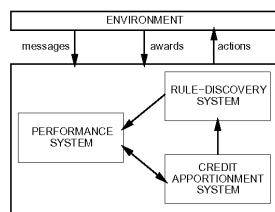$$\text{conditions} \rightarrow \text{action}$$

- example:

$$001\#1 \rightarrow 11010$$

- two approaches:
  - Michingan (Holland): 1 chromosome = 1 rule
  - Pittsburg (De Jong): 1 chromosome = 1 rule set

Evolutionary Computing   Learning classifier systems   2

Learning classifier systems

Basic scheme of classifier systems

Evolutionary Computing   Learning classifier systems   3

Learning classifier systems

- parallel rule application
- rule activation $\rightarrow$ put message on message list
- message (on message list) $\rightarrow$
  – activates rule
  – triggers action
- language

< classifier > ::= < condition > : < Message >
< condition > ::= $\{0,1,\#\}^k$
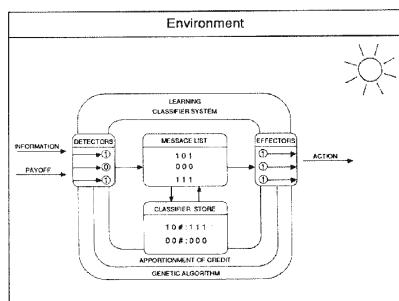< Message > ::= $\{0,1\}^k$

NB. messages form the conditions for (other) rules

Evolutionary Computing   Learning classifier systems   4
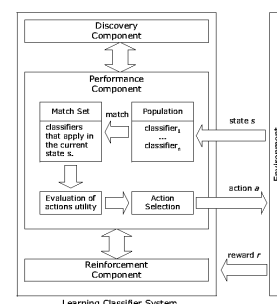
Learning classifier systems

Evolutionary Computing   Learning classifier systems   5

Learning classifier systems

Evolutionary Computing   Learning classifier systems   6

Learning classifier systems

The bucket brigade mechanism (simplified)
- rules have a strength (credit)
- matching rules bid for firing (auction)
- rule with highest bid fires and
  - pays its bid to sender of matched message (clearing)
  - places its message on message board
- after some cycles best rules emerge (useful → strong)

This is used to establish goodness (fitness) of rules

Learning classifier systems

General strength update formula

$$S_i(t + 1) = S_i(t) - B_i(t) - T_i(t) + R_i(t)$$

$B_i = C_{bid} \cdot S_i$ to be payed
$T_i = C_{tax} \cdot S_i$ to be payed
R_i: environmental reward   to be received

C_{bid} and C_{tax} are parameters

Learning classifier systems

		t=0				t=1				t=2			
Index	Classifier	Strength	Messages	Match	Bid	Strength	Messages	Match	Bid	Strength	Messages	Match	Bid
1)	01##:0000	200		E 20		180	0000			220			
2)	00#0:1100	200				200		1 20		180	1100		
3)	11##:1000	200				200				200		2 20	
4)	##00:0001	200				200		1 20		180	0001	2 18	
	Environment	0	0111	20						20			

		t=3				t=4				Final t=5	Payoff
Index	Classifier	Strength	Messages	Match	Bid	Strength	Messages	Match	Bid	Strength	
1)	01##:0000	220				220				220	
2)	00#0:0000	218				208				208	
3)	11##:1000	180	1000			196				206	
4)	##00:0001	162	0001	3 16		156	0001			206	50
	Environment	20				20				20	

Note: 1. C_{bid}, 2. C_{TAX}
Message:=$\{0,1\}^4$ condition:=$\{0,1,\#\}^4$

Learning classifier systems

The Genetic Algorithm
bucket brigade → strength S_i for fitness of rules

T_{GA}: GA period (for GA invocation time)

parent selection: roulette wheel with S_i

offspring creation: standard, but mutation is

$$0 \quad \begin{matrix} 50\% \nearrow 1 \\ 50\% \searrow \# \end{matrix} \qquad 1 \quad \begin{matrix} 50\% \nearrow 0 \\ 50\% \searrow \# \end{matrix} \qquad \# \quad \begin{matrix} 50\% \nearrow 0 \\ 50\% \searrow 1 \end{matrix}$$

survivor selection: a proportion of the population is replaced

offspring's strength - parent's strength
                    - weighted average of parent's strength

Learning classifier systems

SUMMARY

Applications in machine learning tasks (e.g. Boolean mulitplexer, treated later)

Used to be considered as THE genetic approach to this kind of tasks (competition with neural nets)

Genetic programming (after 1992) "outperformed" CS
Revival since the late 90's