Robotics Engineering

Lubna.Z.Bashir*

Received on: 10/2 /2009
Accepted on:7 /1 /2010

Abstract

In this work an approach to robotics engineering called layered evolution and merging features from the subsumption architecture into evolutionary robotics is presented, This approach is used to construct a layered controller for a small simulated robot called street sweeper (SS) that learn crossing a busy street and satisfied its goal of collecting garbage and not getting crushed by passing cars i.e. behaving very adaptively and intelligently in its specified environment. The system uses three classifier systems, which has two levels distributed architecture. Three classifier systems were used to perform complex behavior. First classifier learns simulated robot to crossing the street i.e. move one step toward garbage when there is no predator such as cars passing the street or traffic light color is red . Second classifier learns the simulated robot to stop when the cars are passing the street or traffic light color is green. The third classifier system is controller classifier system should learn switching policy i.e. to which classifier system gives the control when more than one of them is activeTest results, show that the use of a layered evolution can help to control the complexity of learning. a subsumption architecture were design in which each layer is a basic behavior or a control behavior. In addition each single layer have smaller search space therefore the over all task could be easier.

Keywords: Evolutionary robotics, subsumption architecture, Artificial Inelegance, Behavior Based Robotic, Reactive Control.

هندسة الإنسان الآلي

الخلاصة

في هذا العمل قدمنا اسلوب في مجال هندسة الانسان الالي يسمى نشوء الطبقات مع دمج خصائص معمارية المصنفات للسيطرة على سلوك الإنسان الالي. هذا الاسلوب يستخدم لانشاء سيطرة على فعاليات الإنسان الآلي .النظام المفترض (SS)هو إنسان آلي مزيف يسمى كناس الشارع له قابلية تعلم عبور شارع مزدحم بالسيارات ويحقق هدفه في جمع النفايات بدون ان يصدم بالسيارات المارة اي يتصرف بذكاء في بيئته الموصوفة.النظام استخدم ثلاثة مصنفات تعليمية تستخدم لتنفيذ سلوك معقد ، المصنف الأول يعلم الإنسان الآلي عبور الشارع، أي يتحرك خطوة واحدة باتجاه سلة المهملات عندما يكون ضوء إشارة المرور حمراء ولا يوجد سيارات تعبر الشارع.المصنف الثاني يعلم الإنسان الآلي أن يتوقف عندما تكون السيارات مارة أو لون إشارة المرور خضراء ،المصنف الثالث هو مصنف سيطرة لتنسيق فعاليات الإنسان الآلي.نتائج الاختبار بينت ان استخدام أسلوب الطبقات يساعد في السيطرة على تعقيدات عملية تعليم الإنسان الآلي حيث أن معمارية الطبقات تصمم بحيث كل طبقة تمثل سلوك أساسي أو سلوك سيطرة.إضافة إلى أن كل طبقة تملك مساحة بحث صغيرة ولهذا ستكون عملية تعليم الإنسان الآلي ابسط.

* Building and Constriction Engineering Department, University of Technology /Baghdad

https://doi.org/10.30684/etj.28.7.17
2412-0758/University of Technology-Iraq, Baghdad, Iraq

Notations:

AI	Artificial Inelegance.
AOC	Apportionment of Credit.
BBA	Bucket Brigade Algorithm.
ER	Evolutionary Robotics.
GA	Genetic Algorithm.
LCS	Learning Classifier System.
RD	Rule Discovery.
SA	Subsumption Architecture.

1.Evolutionary Robotics

Evolutionary robotics is a relatively novel approach to the relatively mature field robotics, which itself is quite recent compared to the age-old dream of building intelligent machines. A plethora of methods have been used, but the most common way of doing ER (as evolutionary robotics) is this: a set of artificial genomes, called the population, is created. Each genome is a data structure specifying the configuration of a robot controller, and sometimes robot morphology as well. Usually the robot controller is a neural network connected to the sensors and motors of the robot, and the information in the genome is used to specify the synaptic (inter-neuron connection) weights of the network. Then, evolution takes place. All genomes are evaluated, which means that robots with genetically specified controllers are tested on a task and genomes are scored according to how well the controllers they specified did on the task. Good genomes are kept, and bad genomes are replaced with modifications or combinations of the good genomes, and the process is repeated a number of times or until good enough performance has been reached **[10].**

2.Behavior-Based Robotics and The Subsumption Architecture

Evolutionary robotics is certainly not the only modern approach to robotics. In the mid-eighties, **[4]** invented the subsumption architecture, and thereby gave birth to the active research field behavior-based robotics **[9]**, which more or less dominates modern academic robotics research. In this field, like in good old-fashioned AI robotics, robots are hard-wired and behaviors are pre-programmed. Here, briefly describe the principles of the classic subsumption architecture (sometimes abbreviated SA).

A subsumption architecture is organized into layers, where each layer is responsible for producing one (or a few) behaviors. The layers have a strict vertical ordering where higher layers always have precedence over lower layers; often, the behaviors controlled by the higher layers are more complex and "cognitive" than those controlled by the lower layers. Importantly, all layers except the bottom layer presuppose the existence of lower layers, but no layer presupposes the existence of higher layer; in other words, if the robot is built from the bottom up, each stage of the robot development is able to operate**[10]**.

3.Behavior Based Control

The control system of the agent is based on behaviors. A behavior is a subsystem that is responsible for one specific coupling between sensors and actuators **Fig.1.** This contrasts sharply with the view of traditional AI where control is typically based on a set of goals, a model of the world and a search procedure. The search procedure tries to find an action sequence that changes the state of the world to the desired Goal State. If such an action sequence is found, it will be executed in the real world. In a behavior-based agent, the goal need not be explicitly represented. Instead, behaviors are selected on an immediate sensory basis in such a way that they are likely to move the agent closer to the goal in the real world. Problems are avoided when they occur. A behavior-based agent can be augmented with explicit goal representations and planning, but such abilities are not part of its primary repertoire **[1]**.

4.Reactive Control of Robot

Reactive control techniques decompose the operation of the robot into ***task-achieving behaviors*** As can be seen in **Fig.2**, these behaviors are arranged into an order of increasing competence. The lowest behavior, avoids objects, has the highest priority and the most advanced behavior, wandering around in this case, and has the lowest priority. These behaviors run in parallel with each other, but if required the higher behaviors can take control of the robot "subsuming" the behaviors beneath it. The lower behaviors, which have a higher priority, are still functioning and can if required, take control of the robot even when subsumed.

Reactive control of a robot allows it to react to unexpected events in real-time, even if a higher behavior is "thinking" about its task. For example, if a robot with the reactive architecture shown in **Fig.2** is chasing prey and sees a predator, it would switch its behavior to avoiding the predator, since there is no point chasing food, if you are going to get eaten in the process. Furthermore, if the robot now sensed an obstacle in its path, it would again switch behavior, this time to that of avoiding obstacles, since the robot cannot run away from the predator if it is trying to push a tree out of its way. Once the obstacle has been successfully avoided, the robot can swap back to running away from the predator **[7].**

5.Control System Using Classifier System

A control system is an interconnection of components forming a system configuration that will provide a desired system response. A network of different Classifier Systems can implement the control system of an agent. The issue of architecture is therefore the problem of designing the network that best fits some various classes of behaviors **[8]**.

6.What Is Classifier System?

Learning Classifier systems are a form of adaptive system introduced by John Holland in the mid1970s as a form of domain - independent learning system. A classifier system is used as a machine learning system that learns syntactically simple string rules (called classifier) to guide its performance in an arbitrary environment .A classifier system consists of three main components as illustrated in **Fig.3:**

- ***Performance System (Rule and Message System).***
- ***Apportionment of Credit System (Bucket Brigade Algorithm).***
- ***A Rule Discovery (the Genetic Algorithm).***

- **The Performance System**

The performance system (also called rule and message system), is a special kind of production system. A production system is a computational scheme that uses rules as its only algorithmic device. The rules generally have the form. **If <condition> then <action>** the meaning of a production rule is that action may be taken (the rule is fired) when the condition is satisfied. The performance system is composed of: ***classifier store, message list, detector, and effecter.***

1. Classifier Store

The classifier store is the system's long term memory. It is made up of a population of classifiers. A classifier is made up of one or more conditions (known as the ***condition part***) and one action (called the ***action part***). The condition part specifies the set of messages to which a classifier is sensitive, and the action part indicates the message it will broadcast or send out when its condition part is satisfied. Thus a classifier list consists of one or more classifiers of the form: $C_1, C_2, ...C_n / a$ Where: $C_1, C_2, ..C_n \{n>=1\}$ are the conditions making up the condition part and $'a'$ is the action part, conditions are connected by AND operator. The'/' indicates a separation between conditions and action. Each C_i is a string of fixed length K over a fixed alphabet. In most practical systems, the string is defined over three alphabets: {1,0, #}. The '#' is don't care (wild card) symbol that can match any of the chosen symbols {0 or 1}.***for example:*** A classifier having a condition represented by:**1#01** Will recognize (match) the following messages: **1101** and **1001**A classifier posts one or more messages onto the message list when it is activated. The action part of a classifier is used to form the message it sends out when it is activated. It is also a string of fixed length K defined over the alphabet {1,0}.

2. Message list

The message list acts as the system's short-term memory and as the medium for communication between classifiers, and the output interface. It is made up of external messages (input observations) and internal messages (messages from classifiers). A message is represented by a string of fixed length K (same length as that for a condition) over the same set of alphabets {1,0} as the action.

3.Input Interface (Detector)

This receives the input messages from the environment and transforms them into fixed length strings to be placed on the message list

4.Output Interface (Effecter)

Messages placed on the message list by classifiers are processed through the output interface in order to communicate with the system's environment **[3,5].**

•*Apportionment of Credit System (AOC)*

The main task of the apportionment of credit algorithm is to classify rules in accordance with their usefulness. In other words, the algorithm works as follows: a time varying real value called ***strength*** is associated to every classifier'C'. At time zero each classifier has the same strength. When an external classifier causes an action on the environment a payoff is generated whose value is dependent on how good the action performed was with respect to the system goal. This reward is then transmitted backward to internal classifiers that caused the external classifier to fire. The backward transmission mechanism causes the strength of the classifiers to change in time and to reflect their relevance to the system performance (with respect to the system goal).It is not possible to keep track of all the paths of activation actually followed by the rule chains. (A rule chain is a set of rules activated in sequence, starting with a rule activated by environmental messages and ending with a rule performing an action on the environment). Because the number of these paths grows exponentially, it is then necessary to have an appropriate algorithm that solves the problem using only local (in time and space) information. Local in time means that the information used at every computational step is coming only from a fixed recent temporal interval. Spatial locality means that changes in a classifier strength are caused only by classifiers directly linked to it; classifiers C_1 and C_2 are linked if the message posted by C_1 matches a condition of C_2.The classical algorithm used for this purpose is the ***Bucket Brigade algorithm (BBA)***.**[5,6].**

•*The Rule Discovery System (RD)*

A complete classifier system needs some means of generating new rules for use in the performance and learning systems. Well known genetic algorithm (GA) techniques have been used as the main source of rule discovery in Classifier System. Genetic algorithms were inspired by natural selection and operate by evolving generations of individuals, which are successively, more fit according to some fitness evaluation function. In traditional classifier systems classifiers strength is taken as a measure of its fitness or utility in rule discovery. In addition to its use in the performance system, the basic operation of a genetic algorithm is summarized as follows:

1. ***Select classifiers for reproduction:*** The probability of a classifier being selected as a parent is based on its strength.
2. ***Apply genetic operators to the new classifiers:*** Copies of the parent classifiers are generated and transformed using genetic operators. The most commonly used operators are crossover, which combines elements of the bit strings of two parents as in sexual reproduction and mutation which is a change

effected probabilistically on some part of the bit string.

3. ***Select classifiers for deletion:*** In order for the population of classifiers to remain within some reasonable size limit existing classifiers must be deleted as new ones are introduced. There are various means of selecting classifiers for deletion for example probabilistically based on an inverse function of the classifiers strength i.e. weaker classifiers are more likely to be deleted **[2,6,11]**.

7.The Street Sweeper: A Proposed System

In this study the system called ***SS (Street Sweeper)*** uses three classifier systems, which has two levels distributed architecture. Three classifier systems were used to perform complex behavior. First classifier learns simulated robot to crossing the street i.e. move one step toward garbage when there is no predator such as cars passing the street or traffic light color is red . Second classifier learns the simulated robot to stop when the cars are passing the street or traffic light color is green. The third classifier system is controller classifier system should learn switching policy i.e. to which classifier system gives the control when more than one of them is active. The objects in environment are as follows: Moving robot, moving cars, and fixed position represented by traffic light, and fixed position represented by garbage. The environment illustrated in **Fig.4**.since the small robot street sweeper can crossing a busy street and satisfied its goal of collecting garbage and not getting crushed by passing cars would be behaving very adaptively and intelligently in its specified environment.

8.Layered Evolution Structure

Layers evolution approach possible is organizing the controller as a subsumption architecture. Each layer consists of a learning classifier system connected a subset of the robot's sensors and actuators. The layers are connected in a simple structure where higher layers can influence or subsume lower layers

The robot Street Sweeper is controlled by more layers, where each layer consists of a learning Classifier system. Communication between layers is restricted to that higher layers can influence lower layers using a hard-coded, task-specific link. The Street Sweeper (SS) system is built of three learning classifier systems. Organized in two level hierarchical architecture, interacting to gather to perform complex behavior, consist of three classifier systems (LCS-CONTROL), (LCS-APPROACH) and (LCS-STOP).The SS structure is shown in **Fig.5.**

9.The Controller (LCS-Control) Development

The Controller ***LCS-Control*** is used as control system to switch between two classifiers after its analysis the environmental messages, which are received from the environment. The Controller ***LCS-*****Control** is used to learn artificial Robot to choose one of the two classifiers, the basic classifier LCS-Approach, that is, moving towards garbage when there is no cars a crossing the street or stop otherwise. The Controller ***(LCS-Control)*** should learn to ***suppress*** the controller LCS-Stop whenever the Approach behavior proposes an action, which represents ***complex behavior***.

9.1Coding (LCS – Control) Conditions

LCS-Control receives 2-bit message from environment mapping it to 4 states from 0 to 3 of two bit only. The two bit represent as fallows:

- First bit represented are cars crossing the street. (1 the cars not crossing the street, 0 the cars are crossing the street).
- Second bit represented traffic light color. (1 traffic light color is red, 0 traffic light color is green).

LCS – Control Conditions has the form and meaning as in **table.1**

9.2Coding (LCS – Control) Actions

LCS – Control has one action consisting of only one bit, LCS – Control actions have the form and meaning as in **Table**.**2**

9.3Representation of (LCS – Control)

Performance system of the Controller ***LCS-Control*** consists of a message list and classifier store. The classifier stores of LCS-Control contain a set of rules called classifiers, which represents the knowledge and controller of the system at execution time. Condition part of classifier consists of **(2 bit),** and action part consists of **(1 bit)**. The size of classifier store for ***LCS*-Control** will be **(4)** Rules and all classifiers have the same strength value at the beginning.

Example:

The representation of the rule "If the cars are not passing the street and traffic light color is red then the robot approach the garbage “:

Cars are not Passing the street	traffic light is red /	Approach garbage
1	**1**	**1**

9.4Executing of system Street Sweeper Code for (LCS – Control)

The whole project was programmed in Pascal language, Executing the Street Sweeper code, the system responds by presenting the initial report for LCS – Control .The classifier system run for 200 iterations, termination with the snapshot report display in **Appendix. A** the correct rules have achieved high strength values, by contrast the bad rules have strength and bid value near zero. The classifier system eliminates the bad rule quickly there by achieving near perfect performance.

Appendix. A

```
============================
=
 Street Sweeper Code for LCS –
Control
============================
=
population parameters
---------------------
number of classifiers    =    10
number of positions      =     2
number of action         =     1
```

```
bid coefficient            =  0.1000
bid spread                 =  0.0750
bidding tax                =  0.0100
existence tax              =  0.0200
generality probability  =  0.5000
bid specificity base       =  0.2500
bid specificity mult.      =  0.1250
edid specificity base      =  0.2500
ebid specificity mult.     =  0.1250

environmental parameters
------------------------
total number of signal  =     2

apportionment of credit parameters
---------------------------------
bucket brigade flag  =    false

reinforcement parameters
-----------------------
 reinforcement reward    =    10.0

Timekeeper parameters
---------------------
Initial iteration            =     0
Initial block                =     0
Report period                =    200
Console report period        =    200
Plot report period           =    200
Genetic algorithm period   =     10

Genetic Algorithm Parameters
----------------------------
Proportion to select/gen =  0.4000
Number to select          =     2
Mutation probability     =  0.0200
Crossover probability    =  1.0000
Crowding factor              =     3
Crowding subpopulation    =     3
snapshot report
 [block: iteration]  -  [0:0]
```

```
current  status
signal     =    00
Decoded signal     =     0
desired output     =      0
classifier output  =      0
environmental message:     00
no.     strength      bid     ebid   M
classifier
-----------------------------------------------
-------
   1    10.00     0.00      0.00        00:[0]
   2    10.00     0.00      0.00        01:[0]
   3    10.00     0.00      0.00        10:[0]
   4    10.00     0.00      0.00        11:[1]
   5    10.00     0.00      0.00        1#:[0]
   6    10.00     0.00      0.00        #1:[0]
   7    10.00     0.00      0.00        #0:[0]
   8    10.00     0.00      0.00        0#:[0]
   9    10.00     0.00      0.00        ##:[0]
  10    10.00     0.00      0.00        ##:[1]
New winner [1]: old winner [1]

    Initial report for Street Sweeper
       System for (LCS – Control)
snapshot report
[block: iteration]  -  [0:200]
current  status
signal     =    10
Decoded signal     =     2
desired output      =      0
classifier output    =      0
environmental message:     10
no.     strength   bid     ebid   M
classifier
-----------------------------------------------
-------
   1    41.29     0.00      0.00
11:[0]
   2    60.93     3.14      3.10   x
10:[0]
   3   124.55      6.23      6.21  x
10:[0]
```

```
  4   32.58    1.68    1.59  x
10:[0]
  5   49.47    0.00    0.00
11:[0]
  6   31.91    0.00    0.00
11:[0]
  7   36.37    1.87    1.86  x
10:[0]
  8   30.36    1.56    1.55  x
10:[0]
  9   36.10    0.00    0.00
11:[0]
 10   67.51    0.00    0.00
11:[1]
New winner [3]: old winner [3]
```

Last report For Street Sweeper System for (LCS – Control)

10.The (LCS-Approach) development

LCS-Approach is used to learn robot crossing the street i.e. move single step to collecting garbage when there is no cars are crossings the street or the traffic light color is red. The movement capability is completely symmetric a long the two axis .The direction of movement is illustrated in **Fig.6**

10.1Coding (LCS – Approach) Conditions

The length of message, which **LCS – Approach** is received, is always, 3 – bit environment message mapping it to eight states from 0 to 7 of three bit only. The meaning of three bit in input message of LCS – Approach determine relative position of garbage from robot. The form and meaning of three bit LCS – Approach is shown in **Table. 3**.

10.2Coding (LCS – Approach) Actions

The desired action should be the same as system input message. Therefore we have eight actions. Action has the form and meaning as in **table.4**.

10.3Representation of (LCS – Approach)

LCS – Approach consists of a condition part of(3bit) representing the position of garbage in the environment and form action part of (3 bit) representing action to be done in the environment the size of classifier store for LCS – Approach will be (8) rules.

Example:

The representation of the rule "if a relative position of a garbage to be sensed from simulated robot is to the north then the action to be taken by simulated robot is moving to the north, and so on.

Position of garbage From the robot / direction moving of robot

0 0 0 0 0 0

10.4Executing of system Street Sweeper Code for (LCS – Approach)

The whole project was programmed in Pascal language, Executing the Street Sweeper code, the system responds by presenting the initial report for LCS – Approach .the classifier system run for 200 iterations, termination with the snapshot report display in **Appendix . B** the correct rules have achieved high strength values, by contrast the bad rules have strength and bid value near zero. The classifier system eliminates the bad rule quickly there by achieving near perfect performance.

11.The Street Sweeper System Cycle

The using of ***SS*** starts with insertion environment messages to the SS system. Each environment message consists of **2–bit**. These environment messages are received from detectors of the controller ***LCS – Control*** transfer them to its performance system and executed them sequentially only one message in each cycle. In the performance system of the controller ***LCS-Control*** is performed matching process for each environment message with the condition part of all classifiers in classifier store. All classifiers that matched with environment message are sent to the AOC system and use reinforcement learning to reward the winner. In AOC system three procedures are called (Auction, clearinghouse, and tax collector), then a system is calls GA to inject new rule which may increase the performance of the system.If the environment message is matched with at least one of the classifiers in classifier store, then the winner classifier action is transferred to the effecter of the controller ***LCS-Control***. In case of existence of predator,(cars crossing the street or traffic light color Is green) the winner classifier action is **'0'**,i.e (LCS – Control) switch toward (LCS - Stop). in case of no predator,(cars not crossing the street or traffic light color is red) winner classifier action **'1'**.i.e LCS – Control switch toward(LCS – Approach). ***LCS-Control*** is checked whether there is predator or not to determine which of two classifiers will work the controller ***LCS-Approach or LCS- Stop***.

Case of no predator, the controller ***LCS–Control*** switch toward the ***LCS–Approach*** the controller ***LCS-Approach*** receives messages, which consist of **3-bit**. Detector of ***LCS-Approach*** transfers the message to the performance system, In the performance system of ***LCS-Approach*** matching process is performed on each message with the condition part of all classifiers in classifier store. If the message matches at least one of the classifiers in classifier store then the winner classifier action is transferred to effecter of the controller ***LCS-Approach***. the action of winner classifier is sent to the environment by the effectors, to determine the direction of movement for the robot. SS cycle illustrated in ***Fig.7.***

Case of existence of predator, ***LCS-Control*** is switch toward ***LCS – Stop,*** The action of robot is stop.

Appendix .B

```
==================================
                            ======
       Street Sweeper Code for LCS –
                            Approach
==================================
                            ======
                 population parameters
          - - - - - - - - - - - - - - - - - - - - -
        number of classifiers  =    16
         numberof positions     =     3
         number of action       =     3
      bid coefficient          =  0.1000
     bid spread               =  0.0750
      bidding tax              =  0.0100
     existence tax            =  0.2000
   generality probability     =  0.5000
     bid specificity base     =  0.2500
bid specificity mult.    =  0.1250
edid specificity base    =  0.2500
ebid specificity mult.   =  0.1250
environmental parameters
- - - - - - - - - - - - - - - - - - - - - - -
total number of signal   =      3
apportionment of credit parameters
- - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - -
bucketbrigadeflag  =    false
reinforcement parameters
- - - - - - - - - - - - - - - - - - - - - - -
reinforcement reward   =   10.0
Timekeeper parameters
- - - - - - - - - - - - - - - - - - - -
Initial iteration            =     0
Initial block                =     0
Report period               =    100
Console report period       =    100
Plot report period          =    100
Genetic algorithm period    =     -1
Genetic Algorithm Parameters
- - - - - - - - - - - - - - - - - - - - - - - - -
- - -
Proportion to select/gen =  0.4000
Number to select          =      3
Mutation probability      =  0.0200
Crossover probability     =  1.0000
Crowding factor            =     3
Crowding subpopulation   =     3
snapshot report
] block: iteration]  -  [0:0[
current  status
signal    =   000
Decoded signal   =    0
desired output   =    0
classifier output =    0
environmental message:    000
no.   strength    bid    ebid  M
classifier
- - - - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - - - -
- - -
 1    10.00     0.00     0.00
000:[000]
 2    10.00     0.00     0.00
001:[001]
 3    10.00     0.00     0.00
010:[010]
 4    10.00     0.00     0.00
011:[011]
 5    10.00     0.00     0.00
100:[100]
 6    10.00     0.00     0.00
101:[101]
 7    10.00     0.00     0.00
110:[110]
 8    10.00     0.00     0.00
111:[111]
 9    10.00     0.00     0.00
###:[000]
10    10.00     0.00     0.00
###:[001]
11    10.00     0.00     0.00
###:[010]
12    10.00     0.00     0.00
###:[011]
13    10.00     0.00     0.00
###:[100]
14    10.00     0.00     0.00
###:[101]
15    10.00     0.00     0.00
###:[110]
16    10.00     0.00     0.00
###:[111]
new winner[1] : old winner[1]
Initial report for Street Sweeper
System for (LCS – Approach)
snapshot report
]block: iteration]  -  [0:100[
current  status
signal    =   011
```

```
                  Decoded signal    =    3
                  desired output    =    3
                  classifier output =    3
            environmental message:     011
            no.   strength   bid    ebid  M
                                   classifier
-------------------------------------------
-------------------------------------------
                                        ---
  1     0.00     0.00     0.00
000:[000]
  2     0.00     0.00     0.00
001:[001]
  3     0.00     0.00     0.00
010:[010]
  4    36.70     2.29     2.24  x
011:[011]
  5     0.00     0.00     0.00
100:[100]
  6     0.00     0.00     0.00
101:[101]
  7     0.00     0.00     0.00
110:[110]
  8     0.00     0.00     0.00
111:[111]
  9     0.00     0.00    -0.02  x
###:[000]
 10     0.00     0.00     0.05  x
###:[001]
 11     0.00     0.00    -0.01  x
###:[010]
 12     0.00     0.00     0.04  x
###:[011]
 13     0.00     0.00    -0.07  x
###:[100]
 14     0.00     0.00     0.04  x
###:[101]
 15     0.00     0.00    -0.03  x
###:[110]
 16     0.00     0.00     0.01  x
###:[111]
new winner[4] : old winner[4]
```

Last report For Street Sweeper System for(LCS –Approach)

12.The (LCS – Stop) development

LCS – Stop is used to learn robot to still in its position if there is a predator such as cars crossing the street or traffic light color is green. It receives its message from the controller (LCS – Control).

13.Conclusions:

- The approach of layered evaluation which suppose to merge central features of the subsumption architecture, into evolutionary robotics, can help to control the complexity of larning.
- layered evolution faster and more reliably produce better solutions to a given problem than the standard approach. because evolutionary robotics researchers have long noted that there can be advantages to dividing up a problem into several parts. This is the rationale behind incremental evolution.
- By dividing one Layer into many, evolution of them can be sped up not only by making them quicker to update, but also by making the evolutionary search space smaller.
- Evolutionary robotics is often practiced as science rather than engineering.

14.Refrences:

[1]Balkenius. C.(1993),"*Natural Intelligence for Autonomous Agents*" Lund University Cognitive Science, Kungshuset, Lundagard, Sweden.

[2]Bhatia. N, Dixit. P, Sood . M (1999). "*GAMBLE: Genetic Algorithm*

Based Machine learning Expert" Indian Institute of Technology.

[3]Borisov. A and Vasilyev. A(2002), "*Learning Classifier Systems in Autonomous Agent Control Tasks*" Decision Support Systems Group Riga Technical University Department of Computer Science Transport and Telecommunication Institute.

[4]Brooks, R. A. (2002), "*Robot: the future of flesh and machines*" London: Penguin.

[5]Bull.Larry,(2004), "Learning Classifier Systems: A Brief Introduction", Faculty of Computing, Engineering & Mathematical Sciences University of the West of England, Bristol BS16 1QY, U.K. Larry.

[6] Jakobsen. Troels,(2004),"Classifier System Abstracts "Aarhus school of business ,Denmark.

[7]Kelly. L. D.(1997),"*The Development of Shared Experience Learning in a Group of Mobile Robots*" Thesis submitted for the Degree of Doctor of Philosophy, Department of Cybernetics, the University of reading.

[8]Oreizy.P, Gorlick. M, Taylor. R N.(1999), "*An Architecture-Based Approach to Self-Adaptive Software*" IEEE intelligent systems.

[9]Murphy, R. R. (2000), *Introduction to AI Robotics*. Cambridge, MA: MIT Press.

[10]Togelius. J. (2003), "*Evolution of The Layers In a Subsumption Architecture Robot Controller*" Dissertation for the Master of Science in Evolutionary and adaptive systems University of Sussex at Brighton, September.

[11] Zhou. Qing Qing and Purvis. Martin(2004) "A Market-Based Rule Learning System" aGuangDong Data Communication Bureau China Telecom 1 Dongyuanheng Rd., Yuexiunan, Guangzhou 510110, China, Department of Information Science, University of Otago, PO Box 56, Dunedin, New Zealand and/or improving the comprehensibility of the rules.

Table (1) Form and meaning of LCS– Control Conditions

The message	Its meaning
00	**The cars are crossing the street, traffic light is green**
01	**The cars are crossing the street, traffic light is red**
10	**The cars are not crossing the street, traffic light is green**
11	**The cars are not crossing the street, traffic light is red**

Table (2) Form and meaning of LCS– Control actions

The action	Its meaning
1	LCS – Control switch toward LCS – Approach
0	LCS – Control switch toward LCS - Stop

Table (3) Form and meaning of LCS– Approach Conditions

The message	Its meaning
0 0 0	Relative position of garbage from robot is to north
0 0 1	Relative position of garbage from robot is to north - east
0 1 0	Relative position of garbage from robot is to east
0 1 1	Relative position of garbage from robot is to south - east
1 0 0	Relative position of garbage from robot is to south
1 0 1	Relative position of garbage from robot is to south - west
1 1 0	Relative position of garbage from robot is to west
1 1 1	Relative position of garbage from robot is to north - west

Table (4) Form and meaning of LCS– Approach actions

The action	Its meaning
0 0 0	Means robot move to the north
0 0 1	Means robot move to the north - east
0 1 0	Means robot move to the east
0 1 1	Means robot move to the south - east
1 0 0	Means robot move to the south
1 0 1	Means robot move to the south – west
1 1 0	Means robot move to the to west
1 1 1	Means robot move to the north – west

Sensors
Control
Actuators

Figure (1) A behavior is defined as connection between sensors and actuators

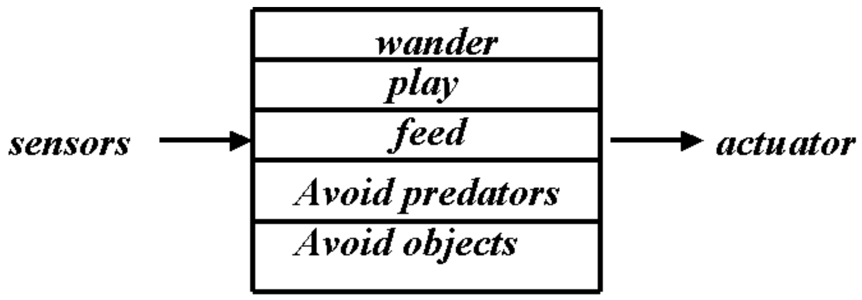

Figure (2) Reactive task decomposition

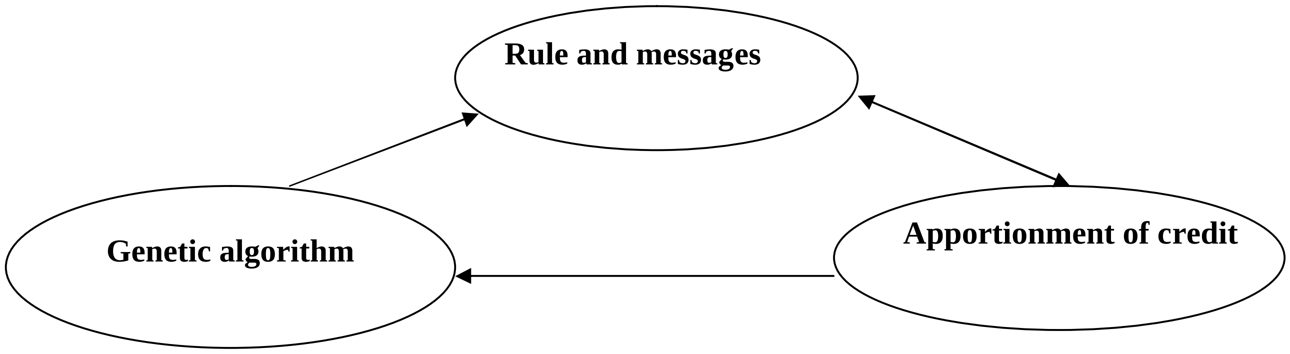

Figure (3) The learning classifier system.

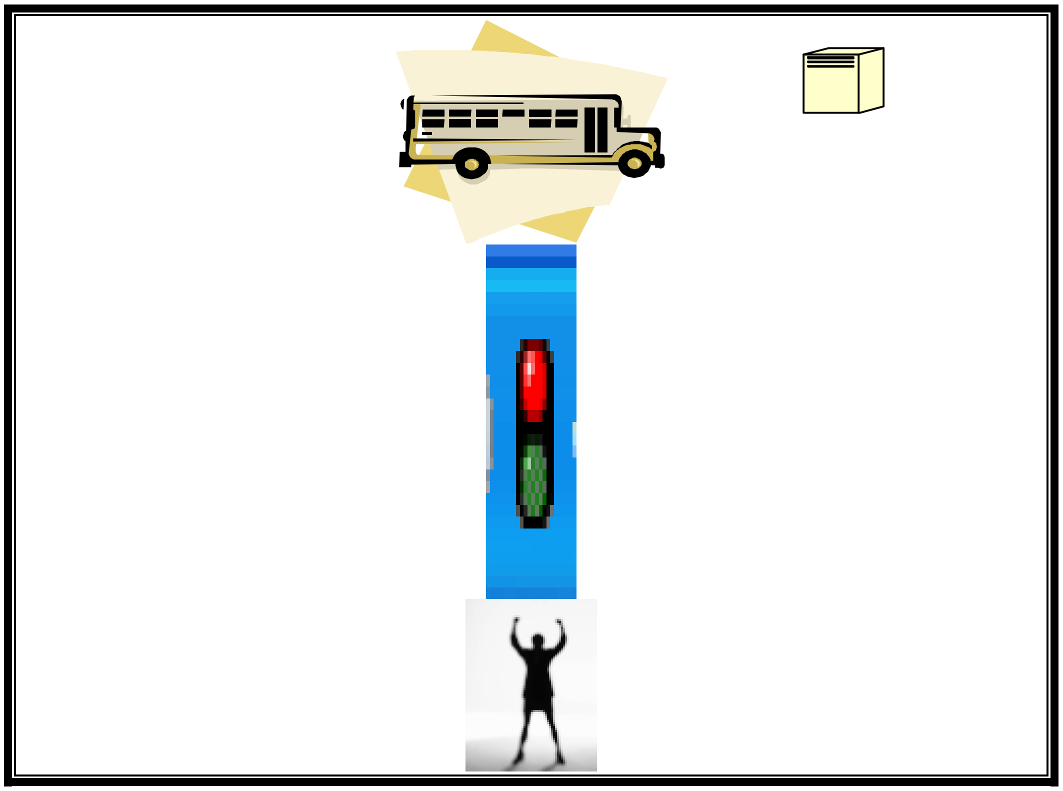

Figure (4) The street sweeper environment.

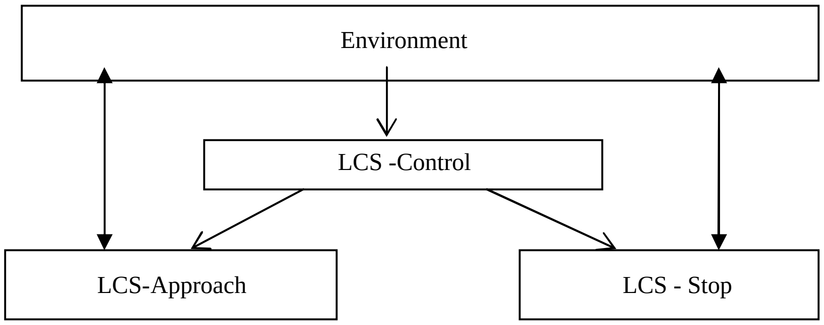

Figure (5) Street Sweeper System Structure.

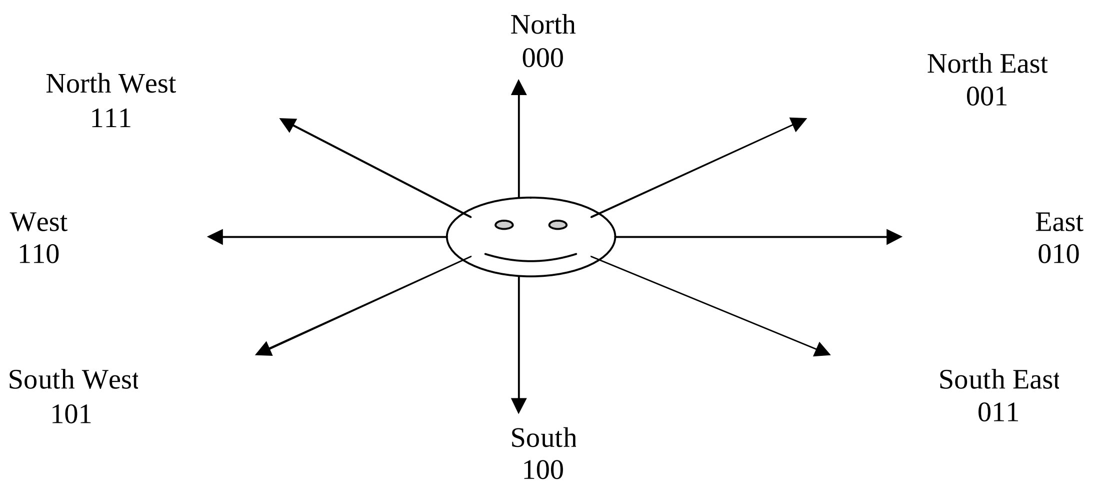

Figure (6) Direction of robot movement

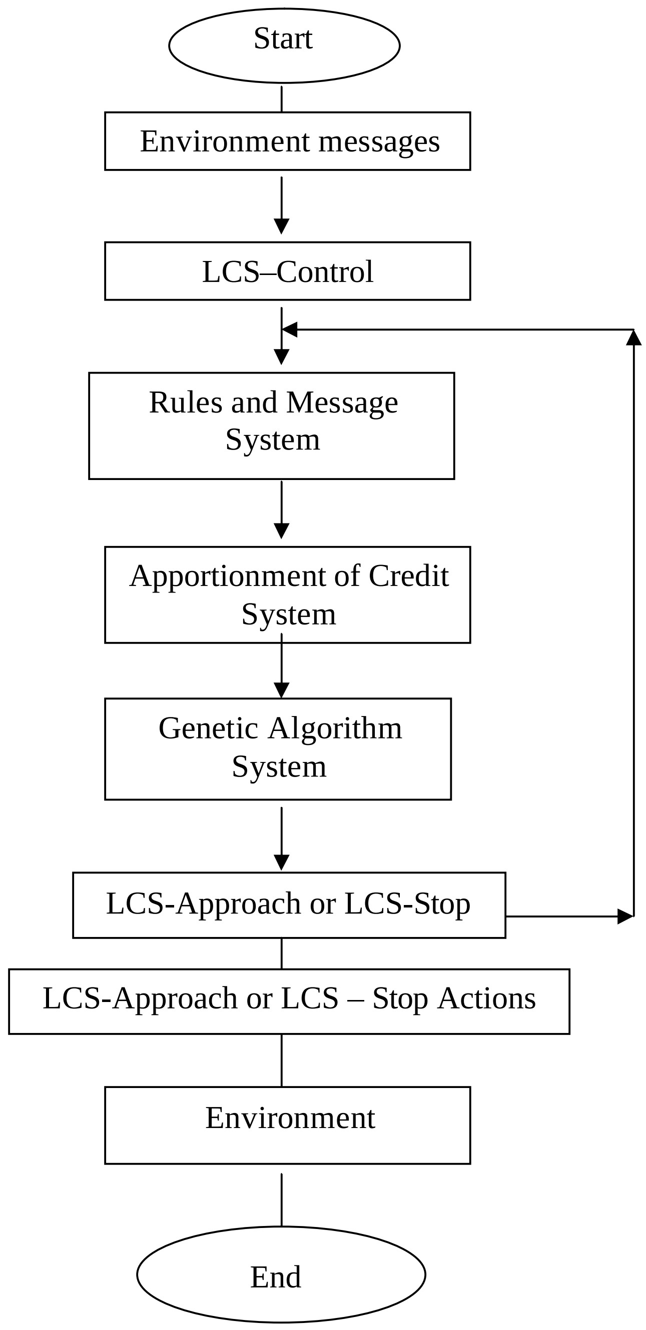

Figure (7) Street Sweeper Cycle