

پردازش تکاملی

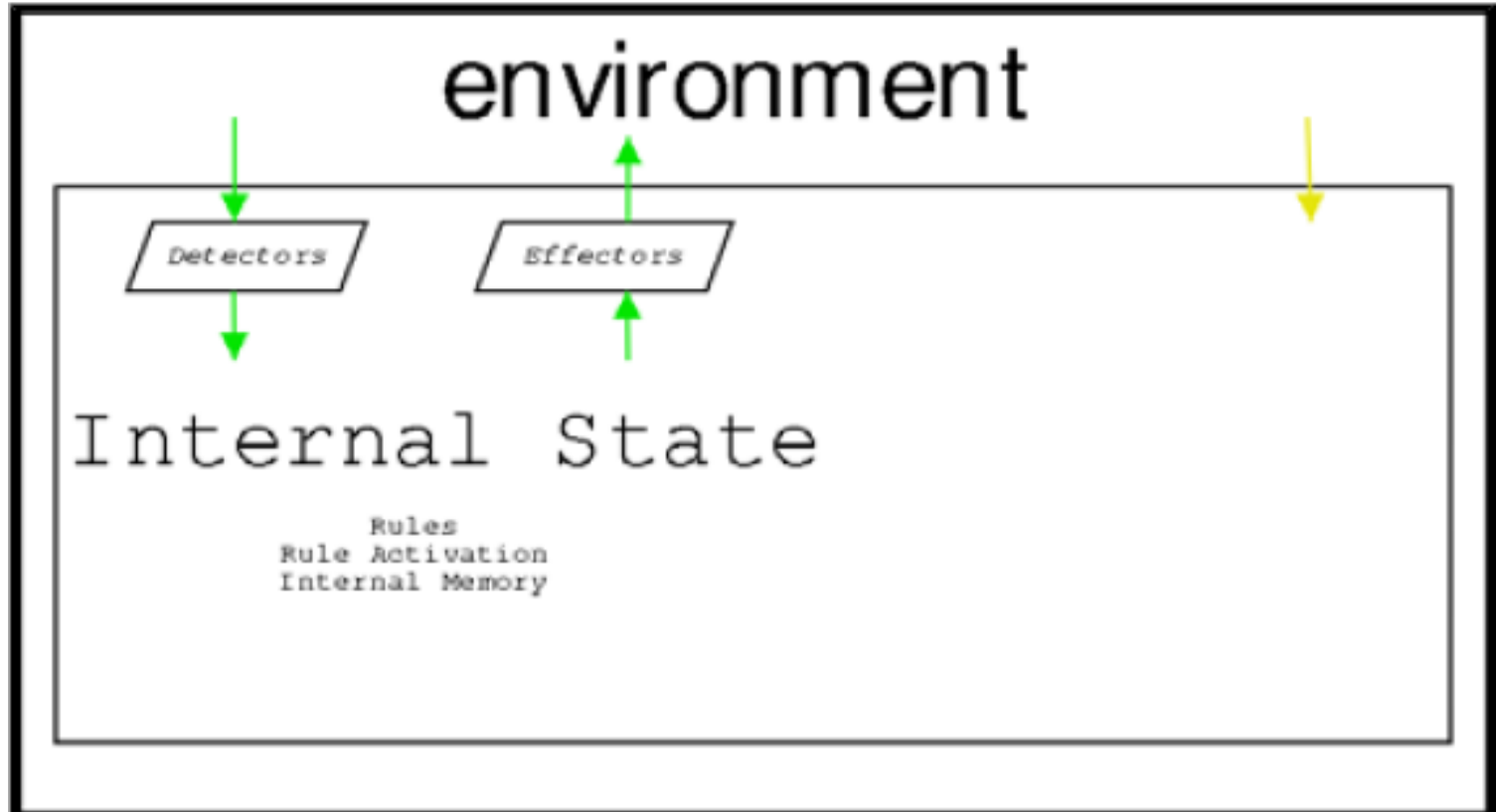
سیستم های طبقه بندی کننده

دانشگاه صنعتی مالک اشتر

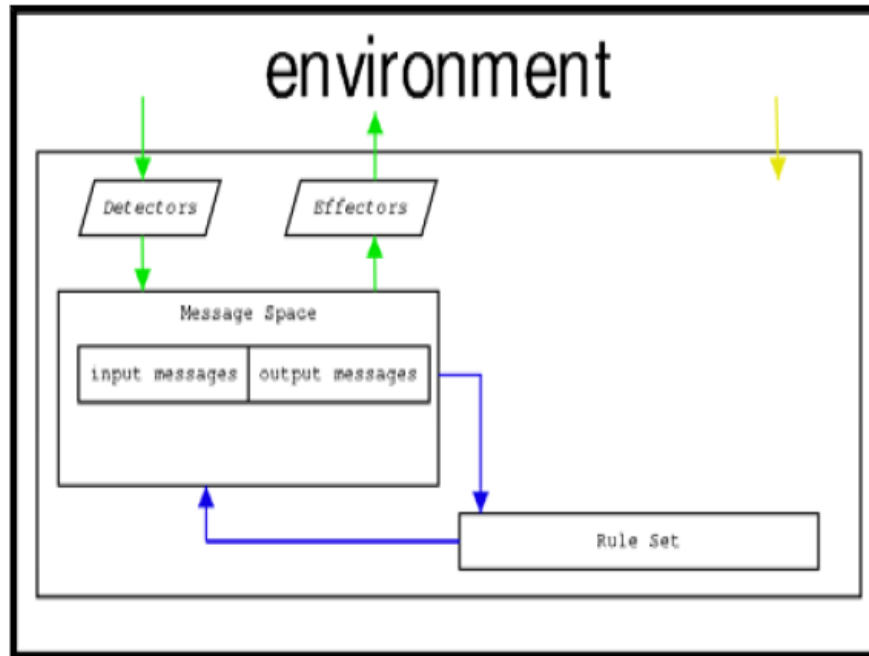
مجتمع دانشگاهی فن آوری اطلاعات و امنیت

زمستان ۱۳۹۲

سیستم های طبقه بندی کننده



CS ها مبتنی بر قاعده هستند



- چرخه اولیه:
 - آشکار ساز پیغام ها را ارسال میکند.
 - قواعد منطبق میشوند.
 - پیغام ها پاک میشوند.
 - قواعد انطباق پیغام ها را ارسال میکنند.
 - فعال کننده ها بروی پیغام ها کار میکنند

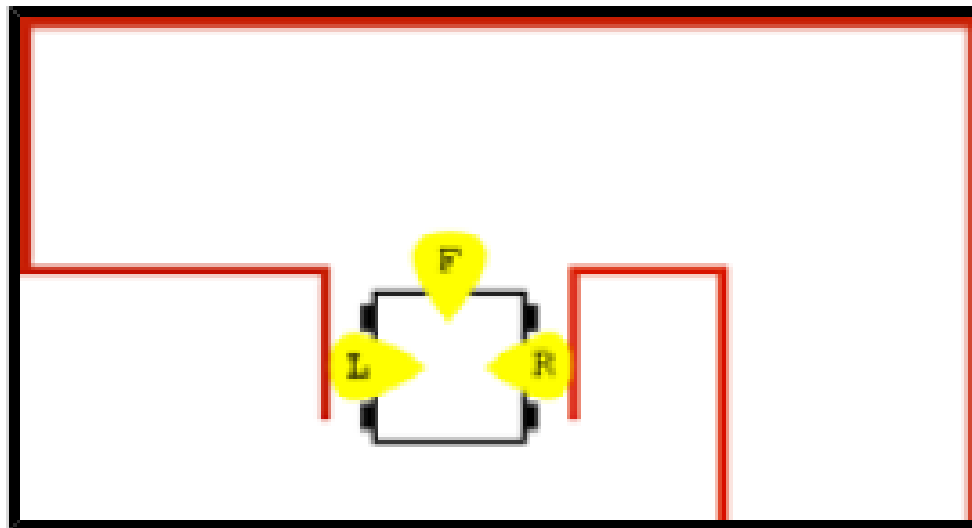
قواعد CS

- قواعد سیستم طبقه بندی کنند:
 - قواعد تولید
 - سمت دست چپ و سمت دست راست
 - $\text{If} \langle \text{condition} \rangle \text{then} \langle \text{action} \rangle$
 - تکمیل محاسبه
 - راه مناسب
- نوشته شده به عنوان رشته ها:
 - پیغام ها: رشته های دودویی
 - سمت دست راست: رشته دودویی
 - سمت چپ: رشته در مبنای سه

نگران سمت دست چپ نباشید

پیغام ناتمام منطبق میشود.

مثال قواعد CS



| Rule | Sensors | | | Actuators | | |
|------|-----------|------------|------------|-----------|------------|------------|
| | wall left | wall front | wall right | turn left | go forward | turn right |
| 1 | # | 0 | # | 0 | 1 | 0 |
| 2 | 0 | 1 | # | 1 | 0 | 0 |
| 3 | 1 | 1 | # | 0 | 0 | 1 |

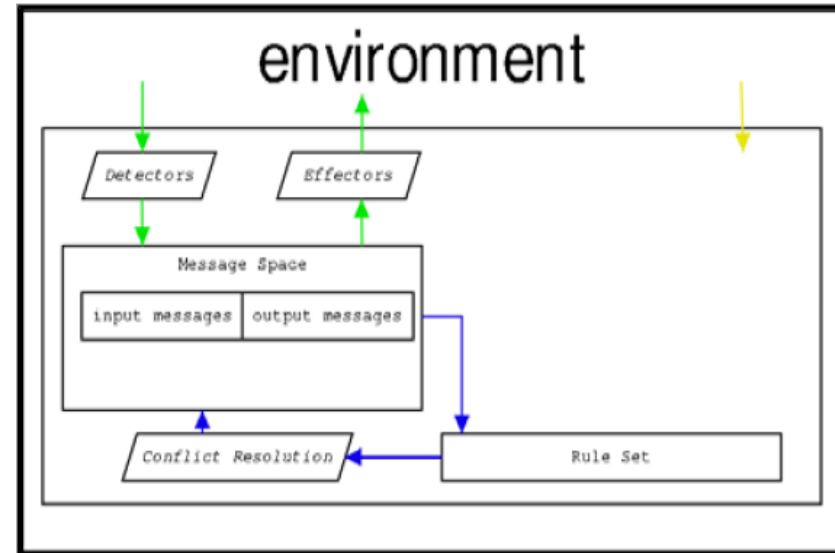
نا سازگاری در قواعد نیست ...!

ناسازگاری قواعد

| Rule | Sensors | | | Actuators | | |
|------|-----------|------------|------------|-----------|------------|------------|
| | wall left | wall front | wall right | turn left | go forward | turn right |
| 1 | # | # | # | 0 | 1 | 0 |
| 2 | # | 1 | # | 1 | 0 | 0 |
| 3 | 1 | 1 | # | 0 | 0 | 1 |

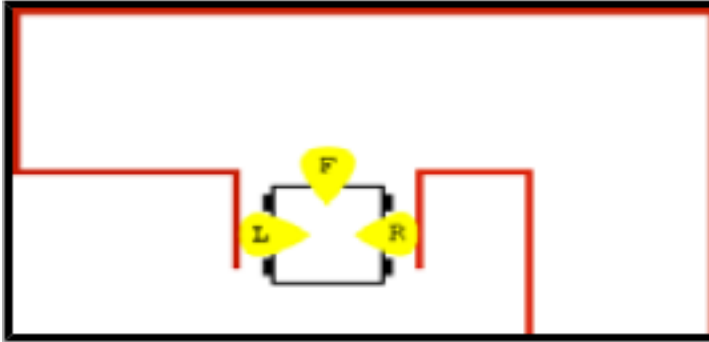
• قواعد ناسازگار:

- بیش از یک قاعده منطبق میشود و
- قواعد متفاوت خروجی های ناسازگار ایجاد میکنند.
- یا اینکه : قواعد بسیار زیادی منطبق میشود.
- صفحه پیغام ها دچار سربار میشود.



نیازی به مکانیسمی میباشد تا ناسازگاری ها را مشخص نمود.

نا سازگاری قواعد

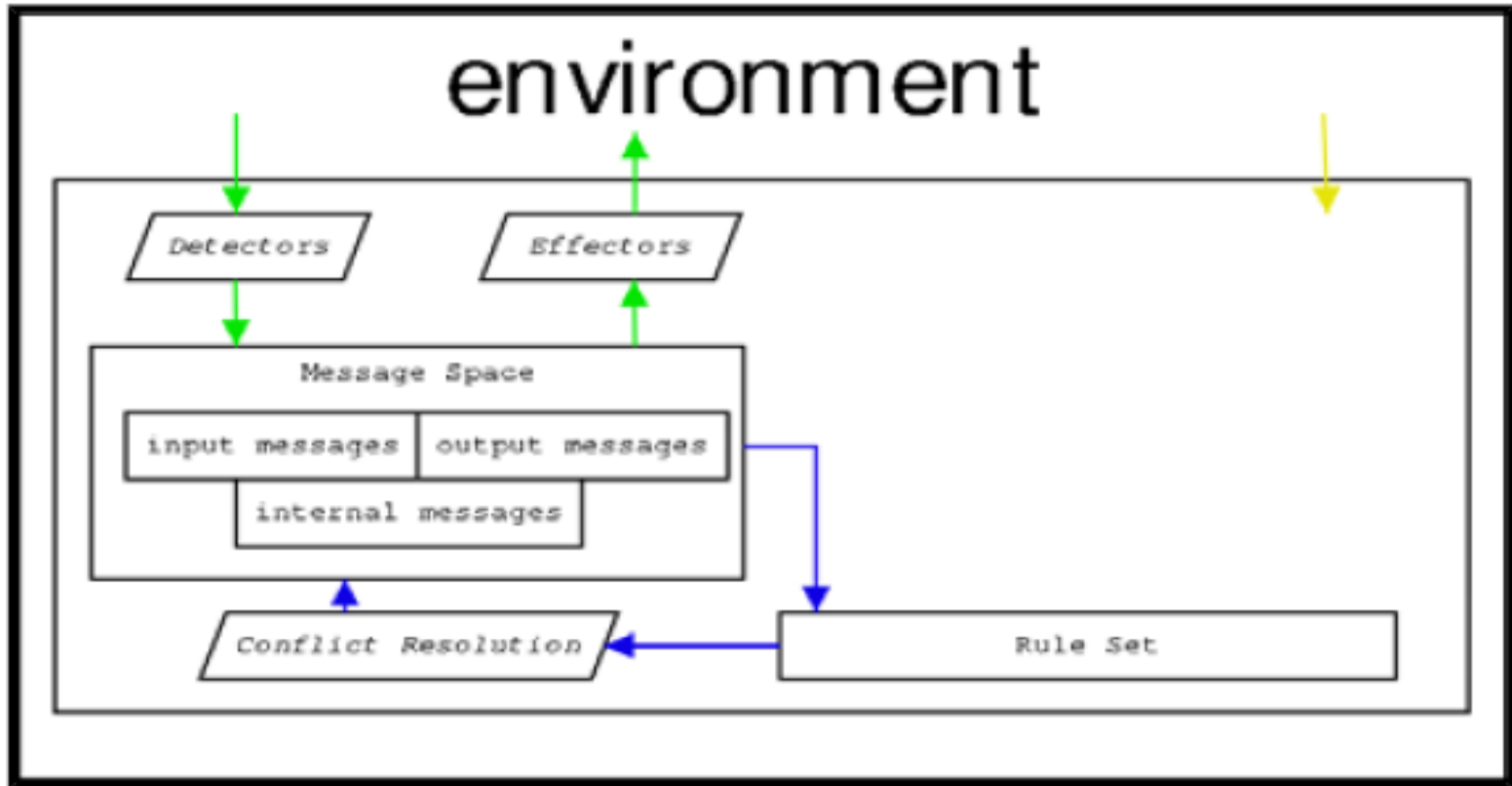


- راه حل : مقادیر قوی
- قوی ترین قواعد پیروز شوند.
- عملکرد همراه با نويز

| Rule | | Sensors | | | Actuators | | |
|--------|----------|-----------|------------|------------|-----------|------------|------------|
| Number | Strength | wall left | wall front | wall right | turn left | go forward | turn right |
| 1 | 10 | # | # | # | 0 | 1 | 0 |
| 2 | 10 | # | 1 | # | 1 | 0 | 0 |
| 3 | 10 | 1 | 1 | # | 0 | 0 | 1 |

پرسش: در مثال بالا کدام مقادیر قوی رفتار صحیح را بدست خواهد داد؟

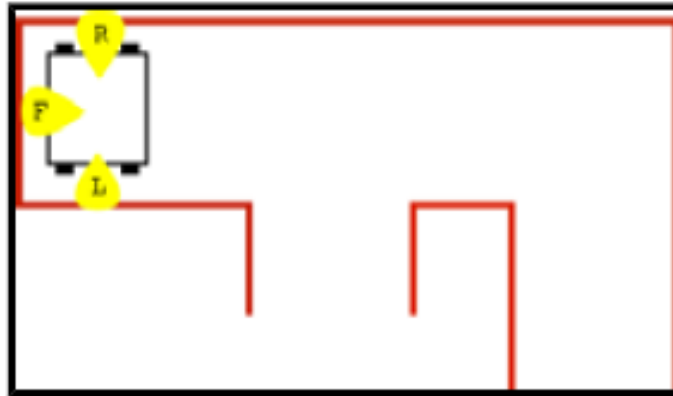
پیغام های داخلی



| Rule | | Sensors | | | Actuators | | | |
|--------|----------|-----------|------------|------------|-----------|------------|------------|---------|
| Number | Strength | wall left | wall front | wall right | turn left | go forward | turn right | go back |
| 1 | 10 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

Evolutionary Computing

مثال: پیغام داخلی (cont)



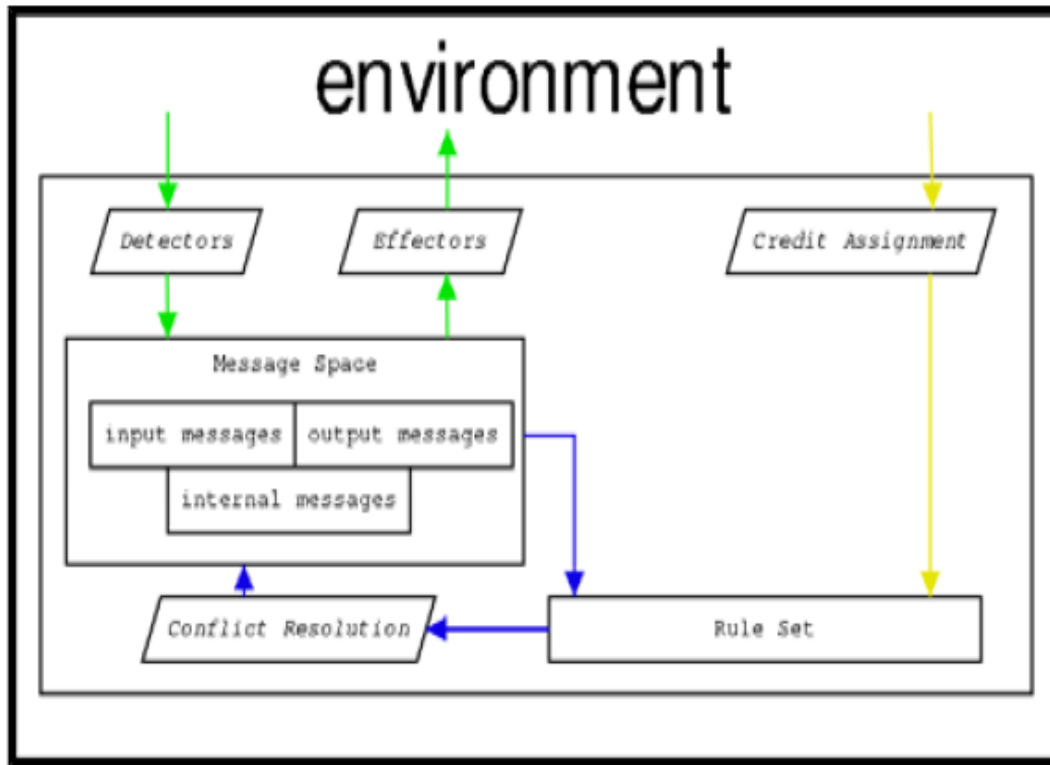
| Rule | | Sensors | | | | | Actuators | | | | | |
|--------|----------|-----------|------------|------------|------------|------------|-----------|------------|------------|---------|------------|------------|
| Number | Strength | wall left | wall front | wall right | internal 1 | internal 2 | turn left | go forward | turn right | go back | internal 1 | internal 2 |
| 1 | 90 | # | # | # | # | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 80 | 1 | # | 1 | 1 | # | 0 | 0 | 0 | 1 | 1 | 0 |
| 3 | 70 | # | # | # | 1 | # | 1 | 0 | 0 | 0 | 0 | 1 |
| 4 | 60 | 1 | 1 | 1 | # | # | 0 | 0 | 0 | 0 | 1 | 0 |
| 5 | 50 | 1 | 1 | # | # | # | 0 | 0 | 1 | 0 | 0 | 0 |
| 6 | 30 | # | 1 | # | # | # | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | 10 | # | # | # | # | # | 0 | 1 | 0 | 0 | 0 | 0 |

حافظه وضعیت داخلی

- LHS پیغام های محیطی و/یا داخلی را منطبق میکند.
- RHS پیغام های محیطی و/یا داخلی را آماده میکند.
- اجازه زنجیره قواعد پیچیده
- اجازه توالی عملکردهای پیچیده

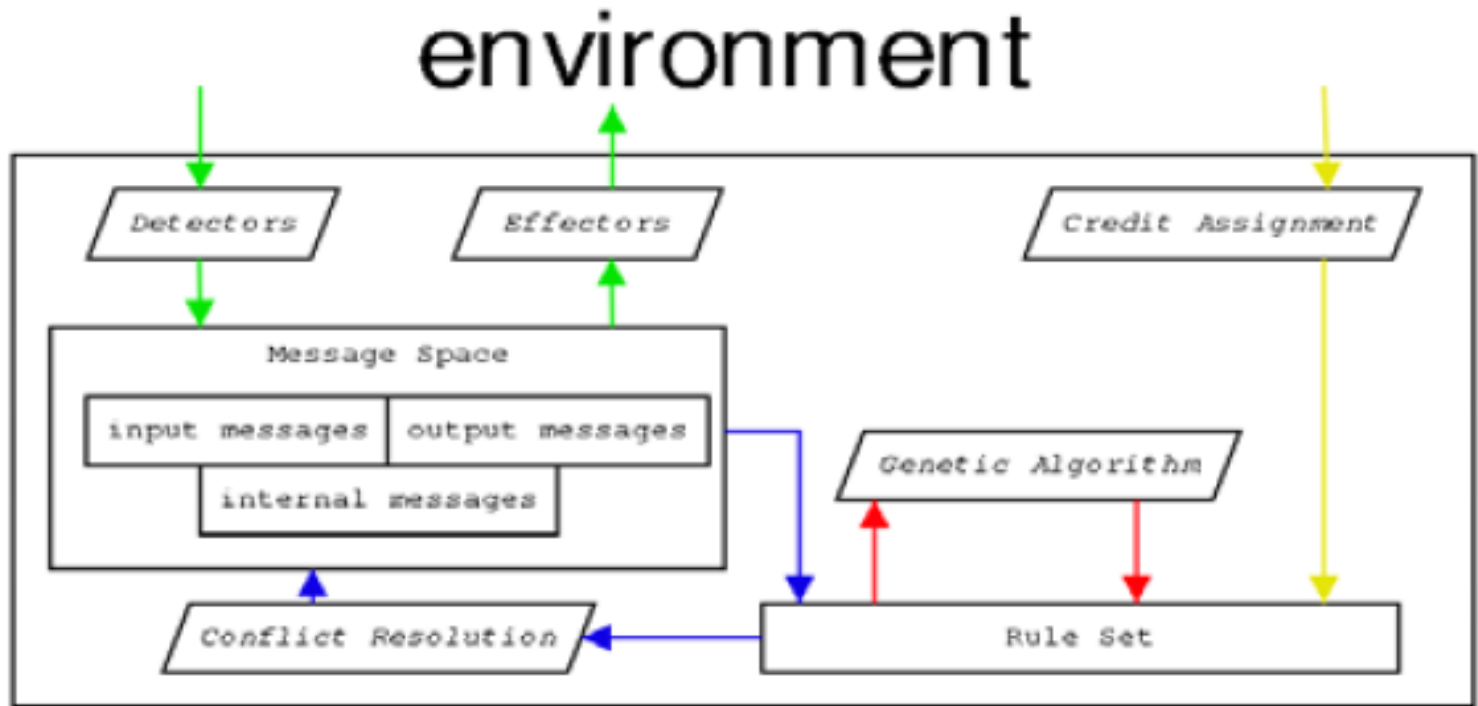
تخصیص اعتبار

نقطه قوت از کجا بدست می آید؟

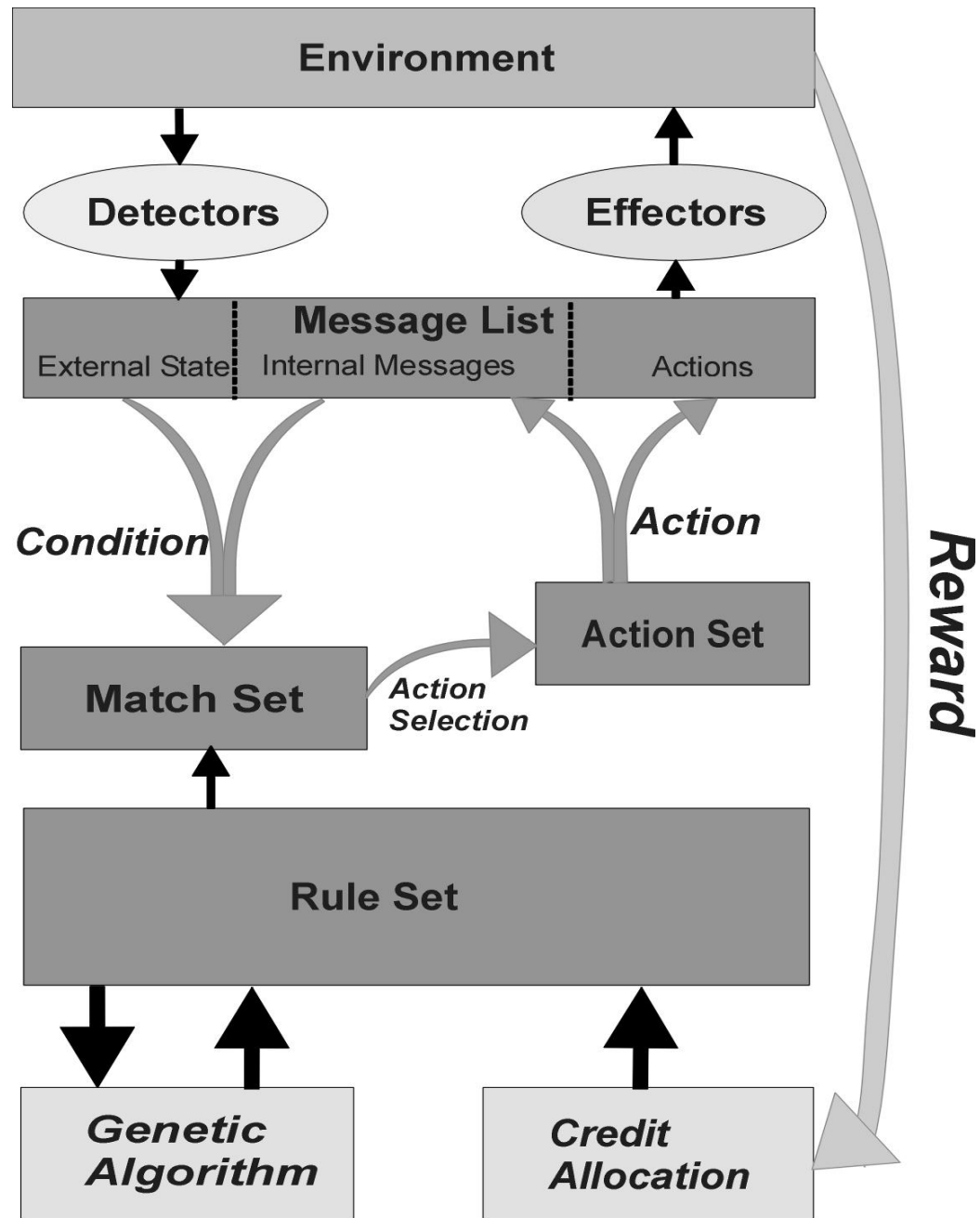


- یادگیری از طریق تعامل
- پاداش از طریق محیط
- مقادیر قواعد به روز
- یادگیری تجدید قوا

گشودن مجموعه قواعد



پرسش: چگونه میتوان مجموعه قواعد را ارائه نمود؟



الگوریتم Bucket brigade

Incremental Strength Adjustment

$$S_{i,t+1} = S_{i,t} - P_{i,t} - T_{i,t} + R_{i,t}$$

$$R_{i,t} = \sum_{k \in K} \frac{P_{k,t+1}}{n_{k,t+1}}$$

• $\text{payoff}_t + \dots$

در جایی K در برگیرنده تمامی طبقه بندی کننده ها خواهد شد که آامین پیغام خروجی در زمان t+1 فعال شده و تطبیق داده میشود.

• and $n_{k,t+1}$ the number of messages matched by classifier k at time t+1

طبقه بندی کننده ها در صورتی تقویت میشوند که:

- فعالیت آنها منجر به بازدهی از محیط شود و یا اینکه
- آنها بخشی از زنجیره فعالیت ها میباشند که منجر به بازدهی خواهد شد.

$P_{i,t}$: Payment

- For being activated
- Bid value, proportional to strength:
 - $P_{i,t} = S_{i,t} * C_{bid}$ if activated
 - $P_{i,t} = 0$ otherwise

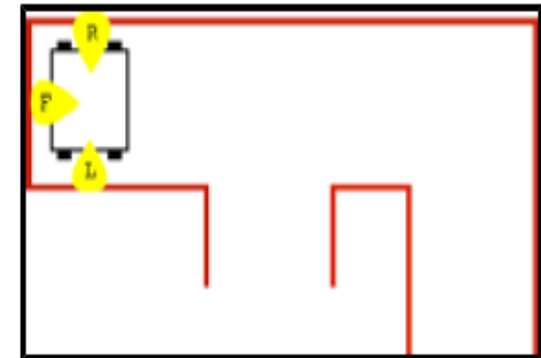
$T_{i,t}$: Tax

- For breathing...
- Proportional to strength:
 - $T_{i,t} = S_{i,t} * C_{tax}$ if activated

اقتصاد اطلاعات

- طبقه بندی کننده آ تامین کننده اطلاعات میباشد.
- طبقه بندی کننده k مصرف کننده اطلاعات است.
- پرداخت "trickle down"

| Rule | | Sensors | | | | | Actuators | | | | | |
|--------|----------|-----------|------------|------------|------------|------------|-----------|------------|------------|---------|------------|------------|
| Number | Strength | wall left | wall front | wall right | internal 1 | internal 2 | turn left | go forward | turn right | go back | internal 1 | internal 2 |
| 1 | 90 | # | # | # | # | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 80 | 1 | # | 1 | 1 | # | 0 | 0 | 0 | 1 | 1 | 0 |
| 3 | 70 | # | # | # | 1 | # | 1 | 0 | 0 | 0 | 0 | 1 |
| 4 | 60 | 1 | 1 | 1 | # | # | 0 | 0 | 0 | 0 | 1 | 0 |
| 5 | 50 | 1 | 1 | # | # | # | 0 | 0 | 1 | 0 | 0 | 0 |
| 6 | 30 | # | 1 | # | # | # | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | 10 | # | # | # | # | # | 0 | 1 | 0 | 0 | 0 | 0 |



t=0

روبات در یک بن بست است.

- Message is 1 1 1 0 0
- Classifier 4 matches and is selected
- Classifier 4 pays 6

t = 1

روبات همچنان در بن بست است.

- Message is 1 1 1 1 0
- Classifier 2 matches and is selected
- Classifier 2 pays 8, this is payed to classifier 4

t=2

روبات دور شده و در یک تقاطع می ایستد

- Message is 0 1 1 1 0
- Classifier 3 matches, and is selected
- Classifier 3 pays 7, this is payed to classifier 2
- Environment pays 3, this is payed to classifier 2

t=3

روبات ۹۰ درجه به سمت چپ چرخش میکند.

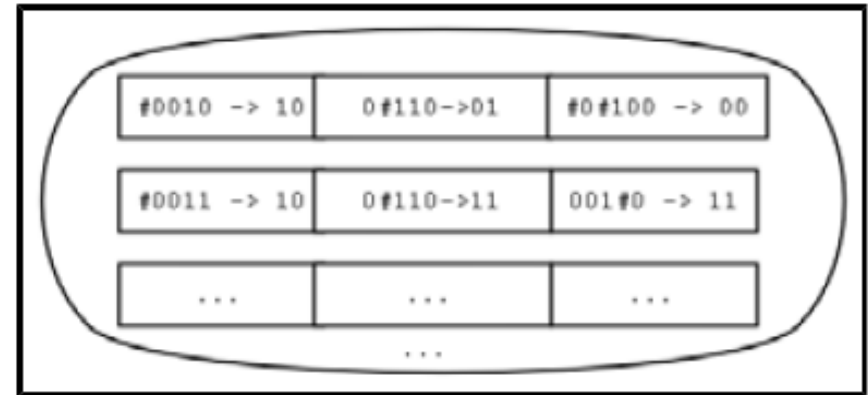
- Message is 0 0 0 0 1
- Classifier 1 matches, and is selected
- Classifier 1 pays 9, this is payed to classifier 3

روش Pitt

De Jong et al, University of Pittsburgh

هر قسمت در برگیرنده مجموعه کاملی از قواعد میباشد

- رشته های طولانی
- بخش های دودویی و سه سه ای
- مجموعه های کوچک از قواعد مناسب



•معایب

- هزینه محاسباتی بالا
- نیاز به زنجیره کامل CS برای هر یک از ارزیابی های تناسب

•مزایا

- سادگی
- تناسب به آسانی تعریف میشود: اجرای CS
- مسائل انحراف خاصی وجود ندارد

عملگرهای اختصاصی در روش Pitt

• در سطح قاعده (Rule)

- قاعده عمومی
- قاعده اختصاصی

• در سطح مجموعه قواعد (Rule set)

- قواعد کپی
- قواعد پاک کردن
- قواعد دو نیم کردن

• در سطح رشته بیت

- اصلاح بیت های شرطی
- اصلاح بیت های عمل

روشی Michigan *Holland et al. University of Michigan*

هر نفر یک قاعده ارائه میکند

- جامعه مجموعه قواعد را می سازد.
- مجموعه های بزرگ از قواعد.

مزیت: سرعت

- تنها یک CS ارزیابی میشود
- ارزیابی بر خط امکان پذیر است

Question: Why is a conventional GA ill-suited?

انحراف در روش Michigan

ما نیاز به انحراف نداریم!

- Convergence => only one type of rule -> bad
- Needed: balanced set of rules

معیارهای جلوگیری کننده از انحراف:

• تکنیک های Niching

• تناسب انتخاب شده با دقت

تناسب در روش Michigan

| Input Message | Output Message | Reward |
|---------------|----------------|--------|
| 00 | 0 | 500 |
| 00 | 1 | 0 |
| 01 | 0 | 500 |
| 01 | 1 | 0 |
| 10 | 0 | 0 |
| 10 | 1 | 100 |
| 11 | 0 | 0 |
| 11 | 1 | 100 |

تابع تناسب چیست؟

- Holland uses strength
- Strength is (proportional to) predicted feedback

مسائل همراه با تناسب پایه ای قوی

- Different niches can have different payoff level
- Little pressure for 'complete' maps of the environment
- Problems with over-general classifiers

محاسبه تناسب در XCF

XCF

(Wilson 1995)

تناسب بر پایه خطای پیشگویی

- Remember: Strength is payoff prediction
- Fitness is predicted accuracy of the payoff prediction

مزایا

- مشکلی در رابطه با سطوح بازدهی متفاوت وجود ندارد.
- دلسردی بروی تعمیم
- دلگرمی به واسطه مسیر دهی کامل ورودی
- دلگرمی به واسطه طبقه بندی کننده های عمومی حداکثری
- سیستم طبقه بندی کننده ارتقاء یافته
- تناسب ارتقا یافته
- Niching
- سایر جزئیات

خطای به روز رسانی شده در طول تقویت

- تقویت به منظور انتخاب عملیات به کار میرود (bidding)
- خطا به منظور انتخاب ژنتیک استفاده میشود.

کلاس های طبقه بندی کننده

مثال

| Classifier | Condition | Action | Predicted Payoff | Prediction Error | Accuracy |
|------------|-----------|--------|------------------|------------------|----------|
| a | # # | 1 | 100 | 0.5 | 0.0 |
| b | 0 # | 1 | 200 | 0.0 | 1.0 |
| c | 1 0 | 0 | 100 | 0.0 | 1.0 |

۳ کلاس مختلف

- Over general: طبقه بندی کننده a
- Maximally general: طبقه بندی کننده b
- Suboptimally general: طبقه بندی کننده c

فضای ورودی پوشی داده نشده

- ورودی 1 1 مفقود است.

Niching etc. in XCF

انتخاب والد از مجموعه فعال

- مجموعه فعال: تمامی قواعدی که در این چرخه محرک میباشند
- هر مجموعه فعال یک Niche را میسازد
- اما: برخی Niche ها متداولتر از سایرین میباشند.

حذف

- ابتدا با دقت پایین
- متناسب با میانگین اندازه مجموعه عمل

Niching etc. in XCF

پوشش:

• زمانی که مجموعه مطابقت خالی شود (هیچ قاعده ای در چرخه انطباق وجود نداشته باشد)

• یک طبقه بندی کننده جدید ساخته میشود.

• LHS: پیغام فعلی

• RHS: تصادقی

رده بندی حذف:

• تمامی طبقه بندی کننده های جدید با آنهایی که موجود میباشند مقایسه میشوند.

• طبقه بندی کننده جدید حذف خواهد شد در صورتی که یکی از آنها هم اکنون موجود باشد:

• صحیح و دقیق است

• آزمون شده است

• منطقاً طبقه بندی کننده جدید را رده بندی میکند

Accurate parents can only have more general offspring

قاعده ارائه پیشرفته

ارائه به صورت رشته بیت ممکن است بهینه نباشد
رشته بیت مبتنی بر And است.

- if $(a = 1) \wedge (b = 0) \wedge (c = 0)$: possible in one rule
- if $(a = 1) \wedge ((b = 0) \vee (c = 0))$: needs two rules
- if (at least three input sensors are 1): needs a number of rules

Real-valued problems are not allowed

- if $(battery < 0.1)$ not possible at all

پرسش: آیا ارائه مناسبتری در وجود دارد؟

شرط های مبتنی بر عبارات S
همانند برنامه ریزی ژنتیک

- Terminal set are the input variables
 - Function set as required
- Output is true (match) or false (no match)
 - ->Very flexible

ساختار سلسله مراتبی در CS

Macaque Visual Cortex:

- 32 Maps
- in 14 layers
- 305 Links spanning max. 7 layers, usually 1-2
 - only 4 areas have 'external' connections

Hierarchie allows higher-level processing

- Abstraction - bottom-up learning
- Decomposition - divide and conquer
- Reuse - solve problems only once

Internal Messages allow classifier chains

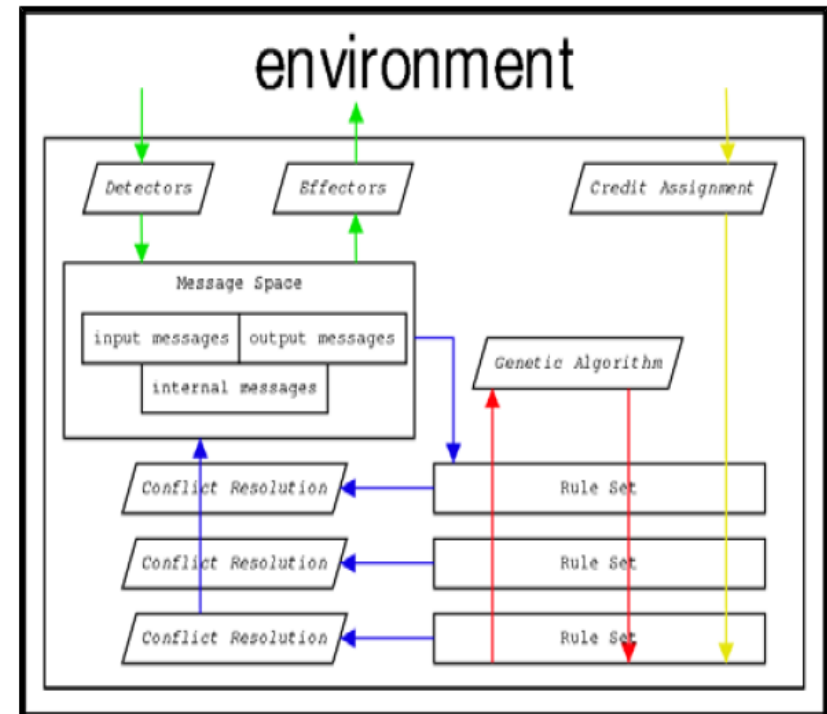
- But: Bucket Brigade not optimal
- Scene-setting rules vulnerable
 - Complex chains rare

Provide Structure - Evolve Hierarchie

- Split system into multiple, interacting CS

Example System:

- N Classifier Systems
- Communicating via shared message board
- Evolved using Pitt approach
- Result: simple task sharing



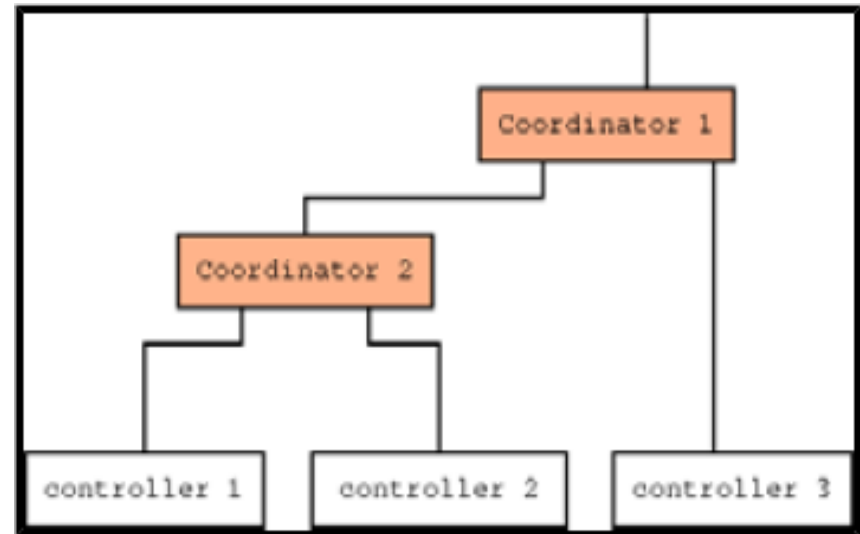
ساختار سلسله مراتبی صریح

Provide Structure and Hierarchie

- Split in multiple CS
- Pre-define hierarchie through message chain

Example System

- *Controllers and Coordinators*
- *Each are independent CS*



(Dorigo and Schnepf 1992)