Generating Stock Trading Signals Based on Matching Degree with Extracted Rules by Genetic Network Programming

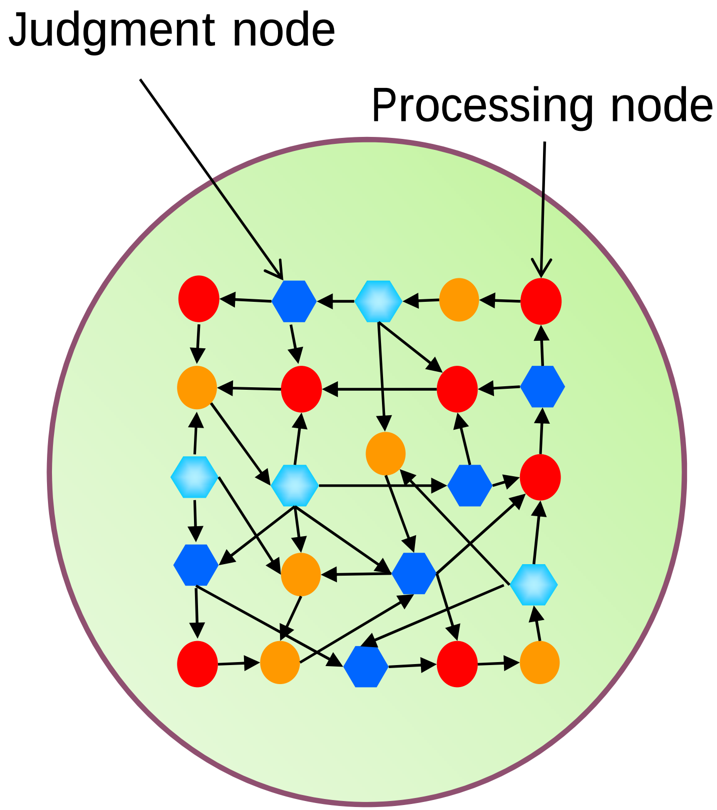
Shingo Mabu, Yuzhu Lian, Yan Chen and Kotaro Hirasawa

Waseda University, Japan

Contents

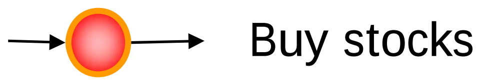
I. Background

- **Genetic Network Programming (GNP)**
  - Graph-based evolutionary algorithm
  - GNP consists of a number of **judgment nodes** and **processing nodes**
  - features
    - Directed graph
    - Compact program
    - Re-usability of nodes

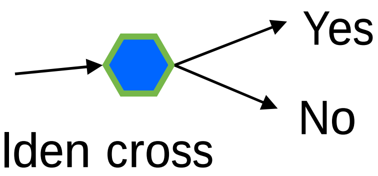


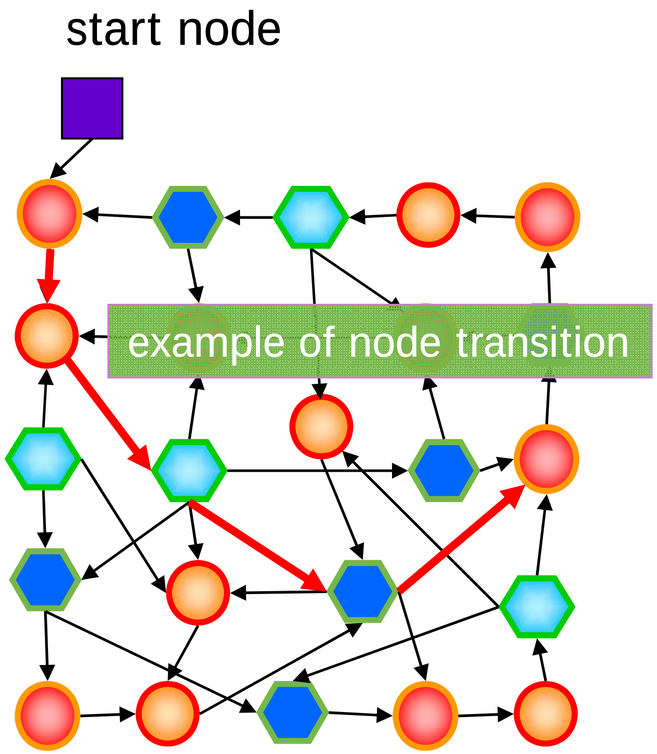
Basic structure of GNP

- Components
  - Processing node
    - Each processing node has the unique action function.

example) 

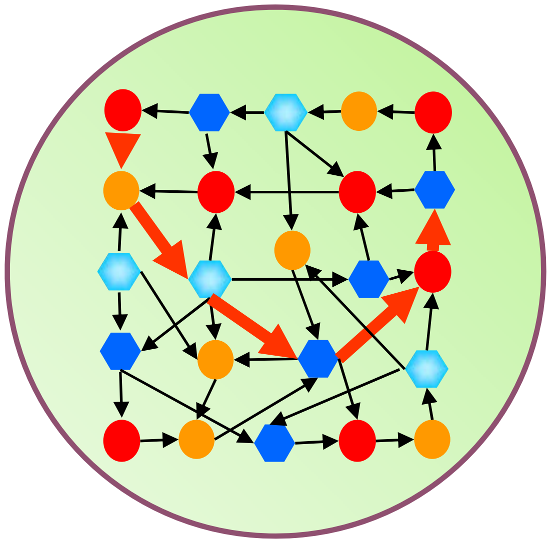
  - Judgment node
    - Each judgment node has the unique judgment function

example) 



I. Background

- **Application of GNP 1: create agent behavior**



Individual**=Program**

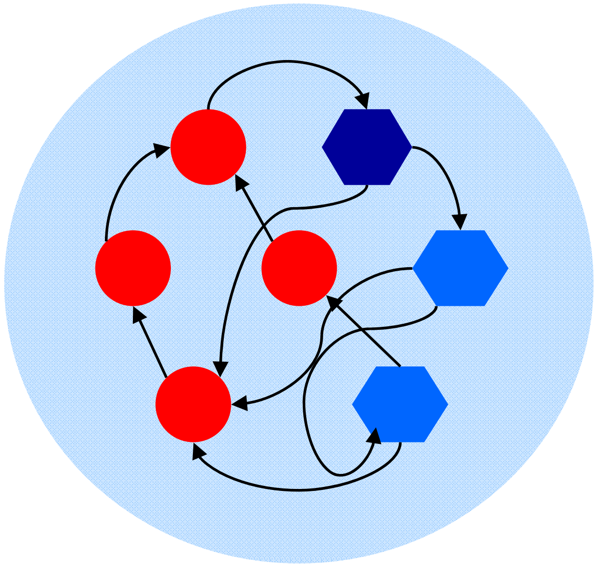
Node transition decides an agent actions

Application

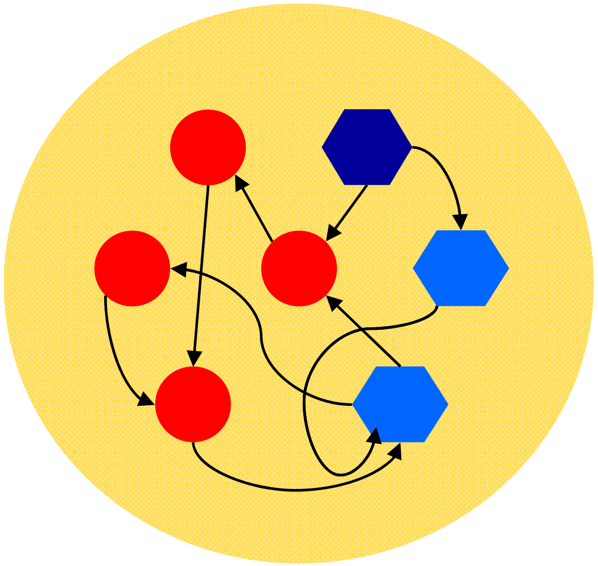
- Robot control
- Stock trading
- Elevator control

I. Background
Evolution (crossover)

- exchange node functions and connections between two parents



Parent 1



parent 2

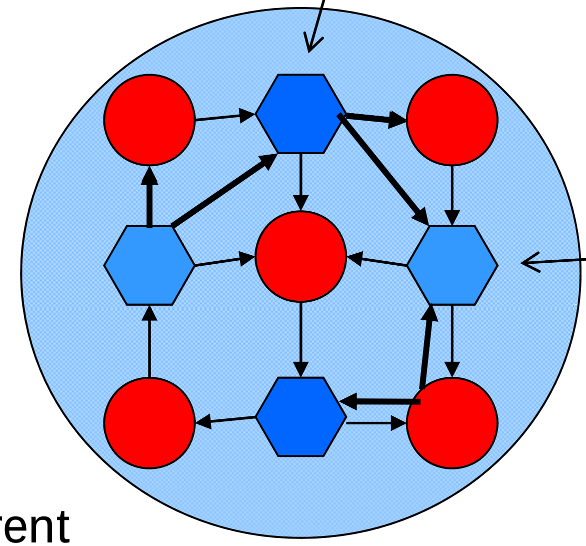
I. Background
Evolution (mutation)

Change node function

Change connections

Judge J_1

Judge J_2



Judge J_3

Judge J_4

parent

I. Background

- Technical analysis on stock markets
  - There are many kinds of technical indexes to indicate buying/selling signals

Technical indexes

- Rate of deviation
- Relative Strength Index (RSI)
- Rate of Change (ROC)
- Volume Ratio
- Stochasitcs
- Relative Correlation Index (RCI)
- Golden/Dead Cross
- MACD
- Candle Chart
- :

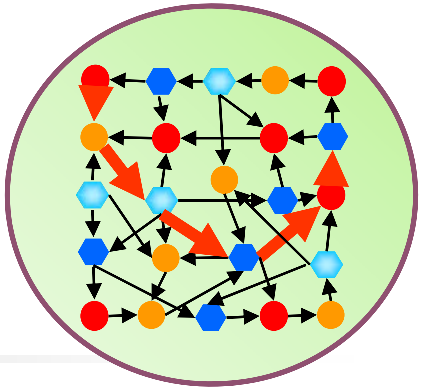
Combine the technical indexes and generate buying/selling rules

Examples of rules

RSI<0.2 & RCI<0.3
→ buy

Stochastics>0.8 & Dead cross=Yes
→ sell

I. Background



- A problem of node transition based GNP
  - It is difficult for one individual to adapt to various kinds of stock market situations

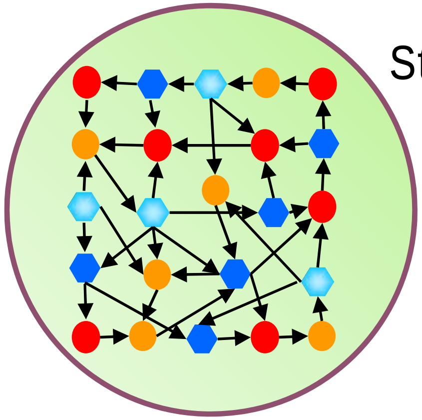
Because...

- Node transition executes judgment and processing nodes one by one.
- All the experiences/rules cannot be used directly to decide the current action.
- The number of rules that one individual can contain is limited due to the limited size of the structure.

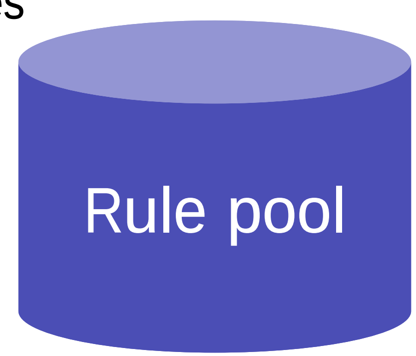
(Large structure causes much computational cost)

I. Background

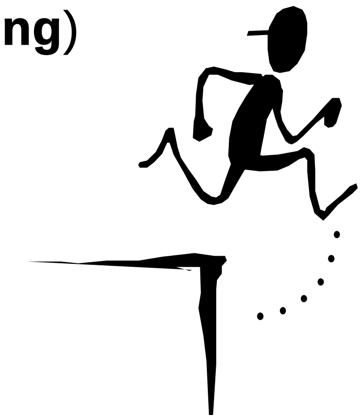
- **Application of GNP 2: Rule Accumulation (GNP-RA)**



Store if-then rules
(**training**)



Decision making
(**testing**)



Individual**=Rule extractor**

Rule pool=solution of GNP

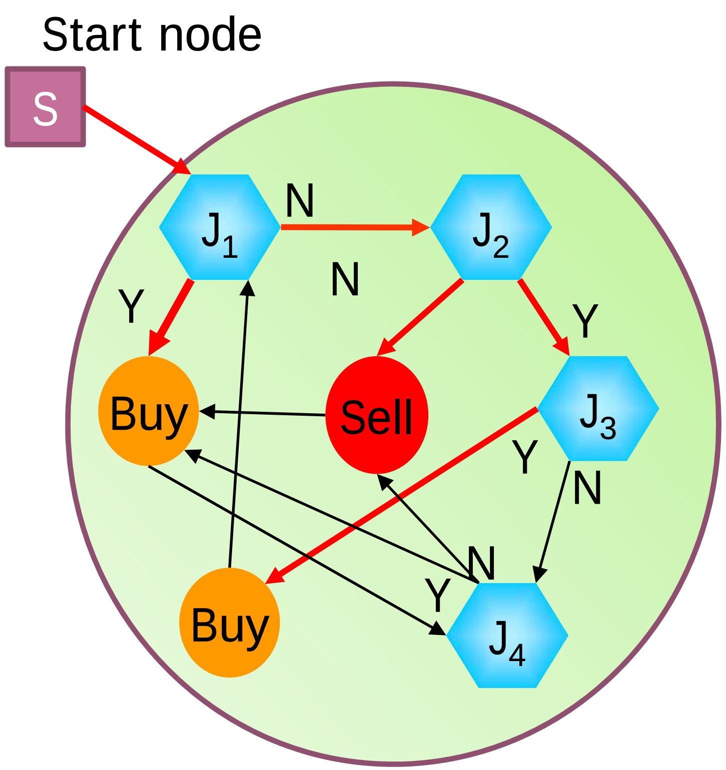
Solution of GNP:
The evolved individuals are not the solution, but the rule pools are the solution.

II. Objective

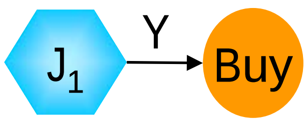
- GNP with Rule Accumulation (GNP-RA) is applied to generate stock trading signals
  - Extract a large number of buying/selling rules
  - Generate buying rule pool and selling rule pool
  - Realize high profitability
- Reinforcement Learning (Sarsa) is used in the rule extraction phase
  - Find more good rules
  - Accelerate the extraction of more rules by its exploration ability

III. Proposed rule extraction algorithm for making stock trading signals

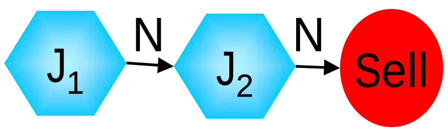
Definition of rule:
A sequence of successive judgment nodes and one processing node



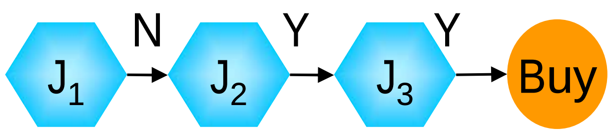
Examples of rules)



$$J_1(\text{yes}) \rightarrow \text{buy}$$



$$J_1(\text{yes}) \wedge J_2(\text{no}) \rightarrow \text{sell}$$

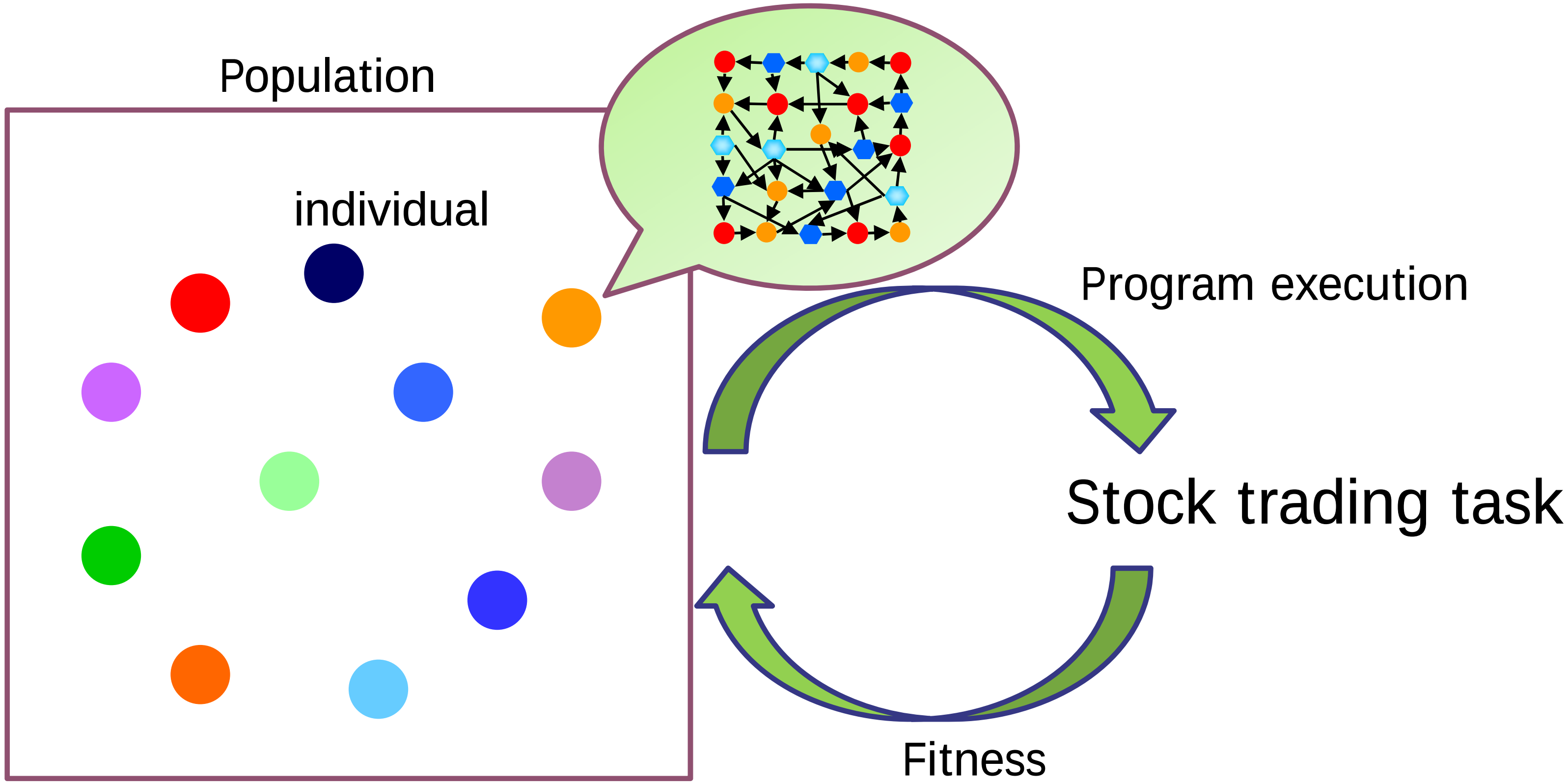


$$J_1(\text{yes}) \wedge J_2(\text{no}) \wedge J_3(\text{yes}) \rightarrow \text{buy}$$

Step 1: Rule extraction in the training phase

Every generation, all the individuals are used as node transition GNP.

→ Execute tasks and calculate fitness values

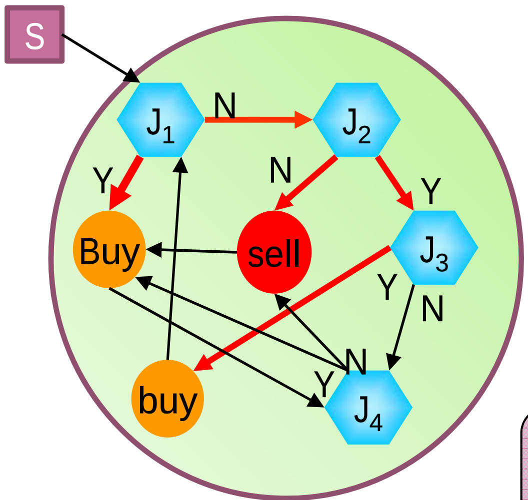


Step 1: Rule extraction in the training phase

Recall the node transition of the elite individual

→ Rules are stored in the rule pools

Start node



Extract good action rules

Buy rule pool

$J_1(\text{yes}) \wedge J_2(\text{yes})$

$J_1(\text{yes}) \wedge J_5(\text{no}) \wedge J_7(\text{no})$

:

Sell rule pool

$J_2(\text{no}) \wedge J_4(\text{yes})$

$J_3(\text{yes}) \wedge J_5(\text{yes}) \wedge J_6(\text{no})$

:

Rate of deviation<0.3?

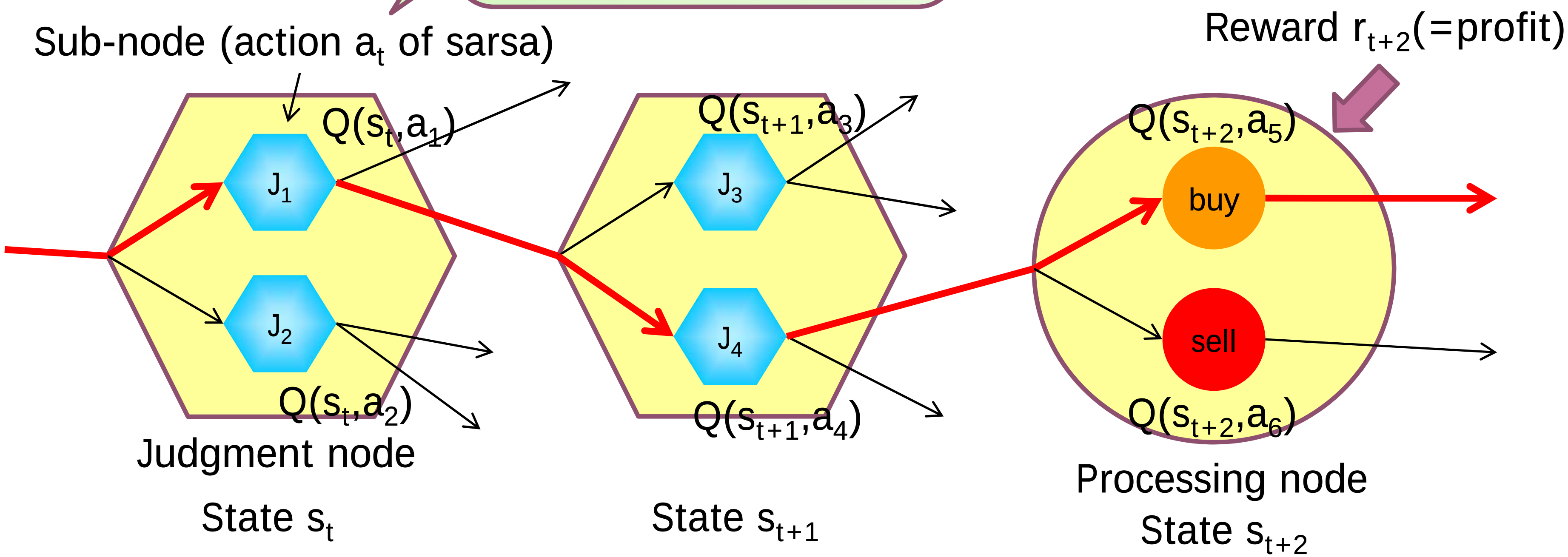
An example of rule: $J_1(\text{yes}) \wedge J_2(\text{yes}) \rightarrow buy$

Golden cross?

Step 1: Rule extraction in the training phase

In order to enhanc ty, **GNP with reinforcement l**

Appropriate sub-node (node function) is selected by Sarsa



GNP with Sarsa optimizes the route of node transition in order to generate better rules.

Step 1: Rule extraction in the training phase

Update equation of Sarsa

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right]$$

Effect of Sarsa in GNP-RA

1. Find the **optimal routes** of node transition
2. explore the graph structures based on ε-greedy policy and **evaluate many candidate rules**

Step 2: Testing phase with extracted rules

- How to decide buying/selling in the testing phase

Definition: Matching degree of stock data *d* with a rule

Data

$$\underline{J_1 = Yes, J_2 = \cancel{No}}, J_3 = Yes$$

$$\mathrm{Match}_{\mathrm{buy}}(\mathrm{d}, \mathrm{rule1}) = \frac{1}{2}$$

$$\mathrm{Match}_{\mathrm{buy}}(\mathrm{d}, \mathrm{rule2}) = \frac{2}{3}$$

rule 1:
$\mathbf{(A_1 = Yes) \wedge (A_2 = Yes) \rightarrow buy}$

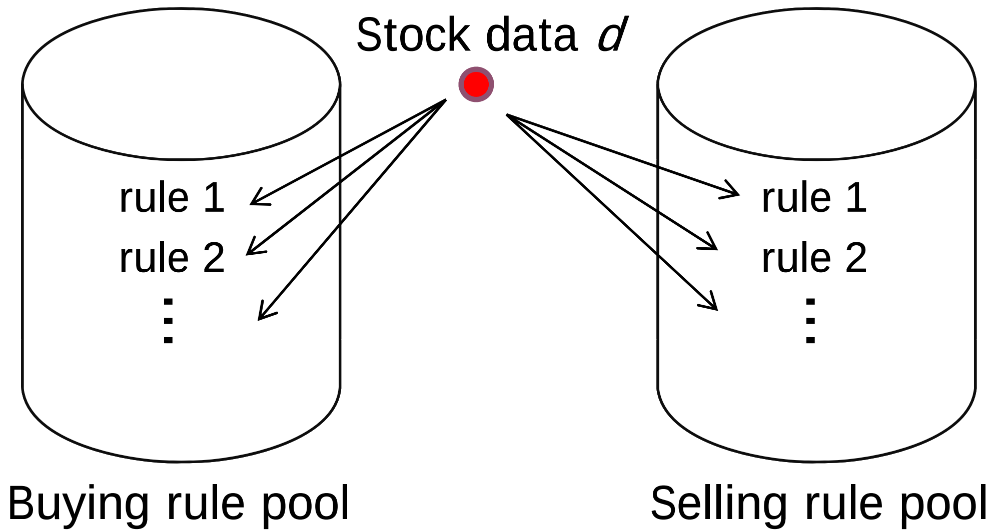
rule 2:
$(J_1 = Yes) \wedge (J_2 = No) \wedge (J_3 = No)$
$\rightarrow buy$

⋮

Buying rule pool

Step 2: Testing phase with extracted rules

Average matching degree



$$\mathrm{m_{buy}(d)} = \frac{\sum_{rule \in R_1} Match(d, rule)}{|R_1|}$$

$$\mathrm{m_{sell}(d)} = \frac{\sum_{rule \in R_2} Match(d, rule)}{|R_2|}$$

R_1: set of buying rules

R_2: set of selling rules

Step 2: Testing phase with extracted rules

Decision of buying/selling

$$m_{buy}(d) \geq m_{sell}(d)$$ ⇒ Buy stocks (if no stocks are hold)

otherwise ⇒ Sell stocks (if stocks are hold)

Average matching degree

IV. Simulations

- Select 16 stocks
  - from the first section of Tokyo exchange
  - One population of GNP is evolved for dealing with one specific stock
    - 16 populations are independently evolved

- Stock price data (training: three years, testing: one year)
  - Case 1: Training (2001-2003), Testing (2004)
  - Case 2: Training (2002-2004), Testing (2005)
  - Case 3: Training (2003-2005), Testing (2006)
  - Case 4: Training (2004-2006), Testing (2007)

- Initial fund: 5,000,000[JPY]
- Fitness = total profit

IV. Simulations

- Node functions

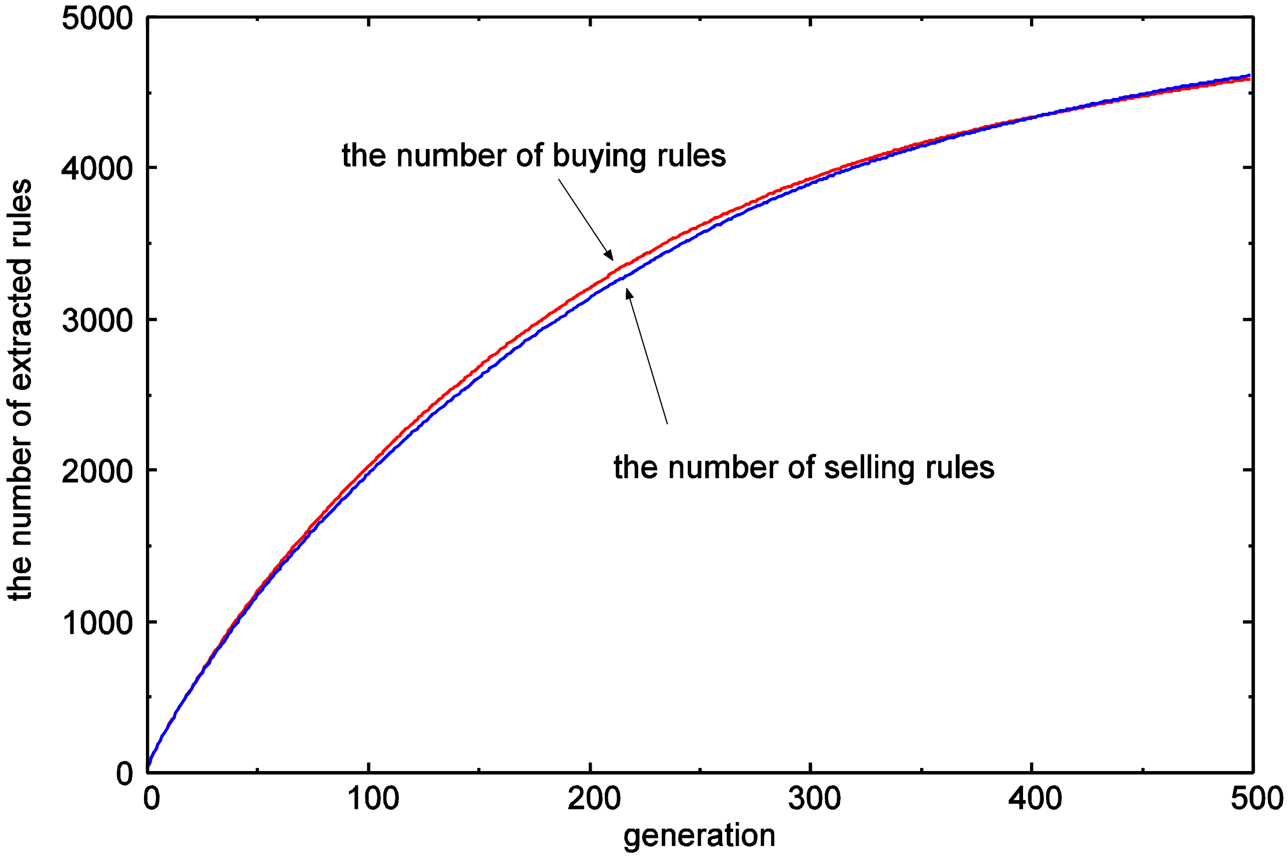
Processing node (2 kind)	Judgment node (21 kinds)
• buying • selling	• Rate of deviation from the moving average (5, 13, 26 days) • RSI (5, 13, 26 days) • Rate of change, ROC (5, 13, 26 days) • Volume ratio (5, 13, 26 days) • Rank Correlation Index, RCI (9, 18, 27 days) • stochastic (5, 13,26 days) • Golden/Dead cross (5 days (short term), 26 days (long term)) • MACD (12days (short term), 26 (long term), 9 (signal)) • candle chart

IV. Simulations

Simulation conditions

- Evolution
  - Number of individuals: 301
  - Number of nodes: 61
    - Judgment node: 40
    - Processing node: 20
    - Start node: 1
  - Crossover rate P_c : 0.1
  - Mutation rate P_m : 0.02
- Sarsa
  - Learning rate α: 0.1
  - Discount rate γ: 0.4
  - ε: 0.1 (ε –greedy policy)

IV. Simulations



The number of extracted rules vs. generation

Profits (testing in 2004

averaged over 30 independent

- Both up and down trend are included in 2004.
- GNP-RA shows larger profits.

stock	GNP-RA (with Sarsa) profit rate[%]	GNP (with Sarsa) profit rate[%]	Buy&Hold profit rate[%]
Toyota	5.5	5.9	10.4
Mitsubishi Estate	23.6	12.0	13.2
Showa Shell Sekiyu	6.4	6.3	6.3
East Japan Railway	6.9	5.1	9.5
NEC	-0.2	-3.0	-20.6
Fuji heavy ind.	-0.6	-0.5	-3.8
Sekisui House	1.6	0.7	5.8
Mitsui & Co.	31.4	18.7	4.8
SONY	-2.0	0.0	3.9
Tokyo Gas	12.2	7.0	7.4
KDDI	9.2	-2.7	-11.6
Tokyo Electric Power	-0.3	1.2	5.2
Daiwa House	-0.7	4.8	0.6
Nomura Holdings	4.6	0.0	-19.8
Shin-Etsu Chemical	11.0	3.9	-5.3
Nippon Steel Coop	9.1	9.6	7.9
mean	7.3	4.3	0.9

Profits (testing in 20

averaged over 30 indepen

- Strong up trend in 2005
- Buy&Hold earns quite large profit.
- GNP cannot hold stocks because GNP sells stocks when the price becomes high.
- GNP-RA shows higher profit than GNP

stock	GNP-RA (with Sarsa) profit rate[%]	GNP (with Sarsa) profit rate[%]	Buy&Hold profit rate[%]
Toyota	28.4	14.9	48.2
Mitsubishi Estate	37.2	36.5	101.1
Showa Shell Sekiyu	37.0	30.0	53.4
East Japan Railway	-0.7	1.3	39.6
NEC	5.4	8.1	14.1
Fuji heavy ind.	4.0	5.9	25.2
Sekisui House	25.7	8.4	26.7
Mitsui & Co.	52.6	38.5	60.4
SONY	1.7	7.4	22.8
Tokyo Gas	13.0	14.1	23.1
KDDI	15.1	18.4	25.0
Tokyo Electric Power	0.0	4.8	13.4
Daiwa House	14.5	14.2	53.5
Nomura Holdings	-0.7	3.9	54.4
Shin-Etsu Chemical	4.3	6.6	47.3
Nippon Steel Coop	43.5	29.6	64.2
mean	17.6	15.2	42.0

Profits (testing in 20

averaged over 30 indepen

- Up trend in 2006
- Buy&Hold earns larger profit.

- GNP-RA shows higher profit than GNP

stock	GNP-RA (with Sarsa) profit rate[%]	GNP (with Sarsa) profit rate[%]	Buy&Hold profit rate[%]
Toyota	17.3	20.8	28.3
Mitsubishi Estate	19.5	26.4	21.6
Showa Shell Sekiyu	-5.6	-5.1	-6.1
East Japan Railway	3.0	-0.4	-4.1
NEC	-24.4	-15.1	-20.7
Fuji heavy ind.	6.2	-3.0	-4.8
Sekisui House	11.2	8.2	15.1
Mitsui & Co.	22.9	11.4	14.0
SONY	0.3	1.0	5.0
Tokyo Gas	5.1	10.5	19.6
KDDI	13.5	17.0	19.6
Tokyo Electric Power	32.6	21.8	35.1
Daiwa House	12.7	8.7	10.4
Nomura Holdings	-5.9	-7.8	-3.8
Shin-Etsu Chemical	14.5	14.6	21.2
Nippon Steel Coop	36.7	46.9	56.9
mean	10.0	9.7	12.9

Profits (testing in 20

averaged over 30 indepen

- Down trend in 2007
- GNP-RA does not make loss on the average
- Buy&Hold always make loss in the down trend

stock	GNP-RA (with Sarsa) profit rate[%]	GNP (with Sarsa) profit rate[%]	Buy&Hold profit rate[%]
Toyota	-16.4	-15.3	-24.4
Mitsubishi Estate	-6.6	-8.8	-9.4
Showa Shell Sekiyu	-3.7	-5.9	-6.2
East Japan Railway	18.2	19.0	15.1
NEC	0.4	0.1	-9.8
Fuji heavy ind.	-5.0	-5.6	-14.3
Sekisui House	-14.2	-24.8	-29.3
Mitsui & Co.	35.9	23.7	23.0
SONY	17.7	18.9	20.5
Tokyo Gas	-10.0	-14.5	-16.1
KDDI	7.3	2.5	2.7
Tokyo Electric Power	-16.5	-19.3	-24.5
Daiwa House	-10.2	-16.3	-24.9
Nomura Holdings	2.1	-10.7	-16.7
Shin-Etsu Chemical	-5.8	-4.9	-12.9
Nippon Steel Coop	8.7	-6.6	1.2
mean	0.1	-4.3	-7.9

Conclusions

- This paper introduces the rule accumulation method by Genetic Network Programming with Sarsa
  - A large number of buying/selling rules are extracted by GNP with Sarsa and stored in the rule pools.
  - Stock trading signals are generated by the matching degree with the extracted rules.
  - Various situations experienced in the training are taken into account to make decision

  - The simulation results in the stock trading simulations show that:
    - the best average profits in the two testing cases out of four is obtained
    - large losses in the down trend do not occur
    - stable results in both trends are obtained

Future work

- There is still room for improvement to judge the situation appropriately.
  - Develop a <u>pruning method</u> to distinguish important and unimportant rules included in the rule pools
  - Develop a method which <u>judges up/down trends</u> (extracts rules of up/down trends)