Postscript: Challenges and Future

Grand Challenges

For computer systems including computer networks, separation of hardware design and software design was accomplished by surpassing the legacy of the von Neumann architecture with stored programs, but the separation is still in progress. For computer systems, cloud computing masks the users and even some programmers from hardware related affairs and troubles. For computer networks too, the software defined network (SDN) has attracted attention. As these trends continue, one important innovation for computer systems will be autonomy and self-management of computer systems. In other words, the high-performance information processing capability of computer systems should be turned on themselves as happens in most biological systems. This is the rationale for studying self-repair networks.

The three major names in this field, John von Neumann, Alan Turing and Norbert Wiener, laid the foundation of computer systems today, and a common inspiration and target of their work was biological systems.

Throughout this book we have often cited von Neumann’s “Theory of self-reproducing automata” as well as “Probabilistic logics and the synthesis of reliable organisms from unreliable components.” If he were to combine these theories with his other pioneering work on “Game theory and economic behavior,” what kind of new science would emerge?

Meanwhile, Turing is famous not only for his Turing Machine, but also the Turing pattern and Turing model in morphogenesis: “The Chemical Basis of Morphogenesis” (Turing 1952).

Wiener is known for “Cybernetics: Control and communication in the animal and the machine” (Wiener 1948), and has had an influence on artificial systems
including computer systems. The concept of homeostasis will become increasingly important for autonomous machines.

Regarding future challenges, if we look at more recent computer scientists and engineers, Jim Gray (Gray 2003) stated twelve Information-Technology Research Goals, among which the following three are consistent with self-repair networks:

9. Trouble-free system;
10. Secure system;
11. Always up.

Tony Hoare listed eight examples “not as recommendation but as examples that may be familiar from the past.” Among the eight examples, “A mathematical model of the evolution of the web” (Hoare 2003) is related to self-repair networks. There is no doubt that self-repair networks as a model alone would not pass “the main tests for maturity and feasibility.” But the topic may be worth pursuing in a more concrete form, for the web exhibits aspects of both a new artificial system and also a natural (collective) system devised by human beings.

**Future Directions**

After money was invented, the economic system of the free market developed its own dynamics and logics, and so cannot always be controlled externally from outside, leading to failures in monetary and financial systems. Similarly, after the Internet was invented, information systems have developed their own dynamics and logics. The Internet is evolving and expanding throughout human society. Although the Internet today may be out of control and difficult to redesign, we need to learn from its emergence as a large-scale information system to help us be more careful in designing and deploying information systems in the future, such as cloud computing systems, global sensor networks and the Internet of Things.

Over the last three decades, computers have become widespread through personalization and interconnection, and have had the following three main impacts:

- Global systemization,
- Local specialization,
- Glocal synchronization

For the autonomy of artificial systems, self-oriented computer systems should be addressed in both technology and theory. Regarding technology, self-controlled computation (focusing on the trade-off of power and accuracy, as well as resilience) has been studied as an extension from Green computing. Regarding the theory, studies on complex systems should focus on the autonomy of artificial systems. Autonomy comes with a huge cost and sophisticated mechanisms, possibly with several hierarchies. One challenge is how to decompose system-level autonomy to atomic-level autonomy or atomic agents. Also, as the mechanism design of game theory has struggled, it is necessary to design and manage an autonomous system
composed of selfish agents to work in a stable fashion within the constraints of efficiency and fairness. Strong AI for managing the autonomous system needs to be revisited to address the fundamental Frame Problem in order for humans to be able to leave self-management tasks to machines that perform as expected, without having to worry that the machines may cause humans unexpected harm.

**A Biological Design of Artificial Systems**

For the design of information systems, we discussed a self-recognizing model for sensor-based systems and self-repair networks for actuator-based systems. Regarding the robustness and resilience of systems, one design principle is to have a uniform and yet developable omnipotent function unit (such as cells for biological systems). Even during the development phase, cells develop while referring to the environment even though the basic structure is framed by genetic information. The function of each component tends to be degenerated (not self-contained completely, requiring further information to be fully specified), and hence individuals reflect information about the environment.

For artificial information systems, it is still difficult to create such an omnipotent unit as cells. However, it is easier to rearrange the components of an information system than a mechanical system, for they are connected electronically rather than mechanically and so physical contact may not be necessary.

As a design paradigm for artificial systems, we have proposed a design that allows diversity in evolving in a dynamic environment learned from the immune system (Ishida 2004).

Simon emphasized the importance of hierarchical systems in design (Simon 1969). For the homeostasis of hierarchical systems, stabilization at each level of hierarchy may be required. For such degenerated and multi-layered stable system, we propose yet another model for the design: a matching automaton, which will rearrange and rematch existing components based on the affinity among them. This design can deal with hardware faults, for new types of threat are emerging such as power viruses and laser attacks that directly cause hardware faults rather than software faults that can be fixed by modifying the states.
Appendix

(Constats for the Steady-State of the Five Repair Types: Mutual AND, Mutual OR, Mixed AND, Mixed OR and Switching AND)


Table A.1 Transition probability in each state transition (infection before repair)

<table>
<thead>
<tr>
<th>State transition</th>
<th>Transition probability (mutual AND-repair)</th>
<th>Transition probability (mutual OR-repair)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(000) → 1</td>
<td>[P_i(1-P_m)(2-P_r+P_rP_m)]</td>
<td>[P_i(1-P_m)(2-P_r-P_rP_m)]</td>
</tr>
<tr>
<td>(001) → 1</td>
<td>[P_i^2(1-P_mP_m)+P_i(1-P_r)((1-P_m)+(1-P_m))]</td>
<td>[-P_i^2(1-P_mP_m)+P_i((1-P_m)+(1-P_m))]</td>
</tr>
<tr>
<td></td>
<td>[P_i(1-P_r)^2]</td>
<td>[P_i^2(1-P_mP_m)]</td>
</tr>
<tr>
<td>(101) → 1</td>
<td>[P_i(1-P_m)(2-P_r+P_rP_m)+(1-P_r)^2]</td>
<td>[P_i(1-P_mP_m)]</td>
</tr>
<tr>
<td>(010) → 1</td>
<td>[-P_iP_rP_m(2(1-P_r)+P_rP_m)]</td>
<td>[(1-P_r)^2+P_i(1-P_m)(2-P_r-P_rP_m)]</td>
</tr>
<tr>
<td>(011) → 1</td>
<td>[-P_i(P_m+P_m)(1-P_r)+P_rP_rP_m]</td>
<td>[-P_iP_rP_m(2(1-P_r)+P_rP_m)]</td>
</tr>
<tr>
<td>(111) → 1</td>
<td>[-P_iP_rP_m(2(1-P_r)+P_rP_m)]</td>
<td>[(1-P_r)^2+P_i(1-P_m)(2-P_r-P_rP_m)]</td>
</tr>
</tbody>
</table>

Table A.2 Coefficients of the equation expressed by parameters of the self-repair network (mutual repair)

<table>
<thead>
<tr>
<th>Constant</th>
<th>Constants expressed by parameters (mutual AND-repair)</th>
<th>Constants expressed by parameters (mutual OR-repair)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>[P_i^2(1-P_r)^2]</td>
<td>[2P_i{1-P_i^2(P_rP_m-1)-P_i(2-P_m-P_rP_m)}]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[P_i^2(1-P_r)^2] [2P_i{1-P_i^2(P_rP_m-1)-P_i(2-P_m-P_rP_m)}]</td>
</tr>
<tr>
<td>B</td>
<td>[-P_i^2(P_m-P_m)^2-P_i(2+P_i)(1-P_r)^2]</td>
<td>[P_i^2(P_m-P_m)^2+4P_i{1-P_i^2(P_rP_m-1)-P_i(2-P_m-P_rP_m)}]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[P_i^2(P_m-P_m)^2+4P_i{1-P_i^2(P_rP_m-1)-P_i(2-P_m-P_rP_m)}]</td>
</tr>
<tr>
<td>C</td>
<td>[-2P_r(1-P_m)(P_r(P_m-P_m)+1)+P_r^2(1+2P_m+P_m)^2+2P_r(1-P_r)^2]</td>
<td>[P_i^2(1+2P_mP_m-2P_r^2)+2P_r(1-P_m-P_m)+2P_i{1-P_r^2(P_m-P_m-1)-P_i(2-P_m-P_rP_m)}]</td>
</tr>
<tr>
<td>D</td>
<td>[P_i(1-P_m)(2-P_r+P_rP_m)]</td>
<td>[P_i(1-P_m)(2-P_r-P_rP_m)]</td>
</tr>
</tbody>
</table>
Table A.3 Transition probability in each state transition where the following conventions are used to accentuate the AND-OR duality (Chap. 7)

<table>
<thead>
<tr>
<th>State transition</th>
<th>Transition probability (mixed AND-repair)</th>
<th>Transition probability (mixed OR-repair)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(000) → 1</td>
<td>$1 - Q_{rn}^3$</td>
<td>$R_{rn}^3 - R_{r}^3$</td>
</tr>
<tr>
<td>(001) → 1</td>
<td>$1 - Q_{rn}^3 Q_{ra} + P_r (Q_{rn}Q_{ra}(Q_{rn} - Q_{ra}) + Q_r)$</td>
<td>$R_{rn}^3 R_{ra}^2 - R_{r}^3 + P_r (R_{rn}^2 - R_{rn} R_{rn}(R_{rn} - R_{ra}))$</td>
</tr>
<tr>
<td>(101) → 1</td>
<td>$1 - Q_{rn}^3 (Q_{ra} + P_r (2 - P_r)) Q_{rn}^2 (Q_{rn} - Q_{ra}) + Q_r$</td>
<td>$R_{rn}^3 R_{ra}^2 - R_{r}^3 + P_r (2 - P_r) (R_{rn}^2 - R_{rn} (R_{rn} - R_{ra}))$</td>
</tr>
<tr>
<td>(010) → 1</td>
<td>$1 - Q_{rn}^3 Q_{ra} + Q_r$</td>
<td>$R_{rn}^3 R_{ra}$</td>
</tr>
<tr>
<td>(011) → 1</td>
<td>$1 - Q_{rn}^3 Q_{ra} + Q_r$</td>
<td>$R_{rn}^2 R_{ra}^2$</td>
</tr>
<tr>
<td>(111) → 1</td>
<td>$1 - Q_{rn}^3 + Q_r$</td>
<td>$R_{rn}^3$</td>
</tr>
</tbody>
</table>
Table A.4 Coefficients of the equation expressed by parameters of the self-repair network (mixed repair) where the following conventions are used to accentuate the AND-OR duality (Chap. 7)

<table>
<thead>
<tr>
<th>Constant</th>
<th>Constants expressed by parameters (mixed AND-repair)</th>
<th>Constants expressed by parameters (mixed OR-repair)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
<td>$(Q_m - Q_{ra})^3 + 2P_i Q_{ra}(Q_m - Q_{ra})^2 + P_i^2(Q_m - Q_{ra})^2 + Q_r$</td>
<td>$- (R_m - R_{ra})^3 - 2P_i R_{ra}(R_m - R_{ra})^2 + P_i^2(R_m^2 - R_{ra}^2(R_m - R_{ra}))$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$3R_{rn}(R_m - R_{ra})^2 - 2P_i R_{ra}(R_m - R_{ra})(2R_m - R_{ra})$ - $P_i^2(R_m^2 - R_{ra}^2(R_m - R_{ra}))$</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>$3Q_m (Q_m - Q_{ra})^2 - 2P_i Q_m(Q_m - Q_{ra})(Q_m - 2Q_{ra}) - Q_r$</td>
<td>$3Q_{rn}(R_m - R_{ra}) + 2P_i Q_{rn}(R_m - R_{ra}) + Q_r$</td>
</tr>
<tr>
<td></td>
<td>$3Q_m^2(Q_m - Q_{ra}) + Q_r - 1$</td>
<td>$- 3R_{rn}^2(R_m - R_{ra}) + R_m^3 - 1 + 2P_i R_{ra}^2 R_{rn}(R_{rn} - R_{ra})$</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>$1 - Q_{rn}^3$</td>
<td>$R_m^3 - R_{rn}^3$</td>
</tr>
</tbody>
</table>

Note: The table entries are derived from the equations presented in the text, using the conventions for the AND-OR duality. Each coefficient or constant is expressed in terms of the parameters $Q_m$, $Q_{ra}$, $R_m$, $R_{ra}$, and $P_i$.
<table>
<thead>
<tr>
<th>State transition</th>
<th>Transition probability (switching AND-repair)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(000) → 1</td>
<td>$P_s (1 - P_{sr}) { {1 - (1 - P_{sr})P_r (1 - P_m) } { P_{sr} + (1 - P_{sr}) (1 - P_{sr}P_r + P_{sr}P_rP_m) } + (1 - P_{sr}) P_r (1 - P_m) }$</td>
</tr>
<tr>
<td>(001) → 1</td>
<td>$P_s { 1 - (1 - P_{sr})P_r (1 - P_m) } { P_{sr} (1 - P_m) + (1 - P_{sr}) (1 - P_{ra}) (1 - P_{sr}P_r + P_{sr}P_rP_m) } + (1 - P_{sr}) P_r (1 - P_m) +$</td>
</tr>
<tr>
<td></td>
<td>$P_s { { (2 - P_t) P_{sr}P_r (1 - P_m) } { 1 - (1 - P_{sr}) P_r (1 - P_m) } { 1 - (1 - P_{sr}P_r + P_{sr}P_rP_m) } } + (1 - P_{sr}) P_r (1 - P_m) +$</td>
</tr>
<tr>
<td>(010) → 1</td>
<td>$P_r { 1 - (1 - P_{sr})P_r (1 - P_m) } { P_{sr} (1 - P_m) + (1 - P_{sr}) (1 - P_{ra}) (1 - P_{sr}P_r + P_{sr}P_rP_m) } + (1 - P_{sr}) P_r (1 - P_m) +$</td>
</tr>
<tr>
<td>(011) → 1</td>
<td>$P_r (1 - P_{ra}) { 1 - (1 - P_{sr})P_r (1 - P_m) } { P_{sr} + (1 - P_{sr}) (1 - P_{sr}P_r + P_{sr}P_rP_m) } +$</td>
</tr>
<tr>
<td>(111) → 1</td>
<td>$P_r (1 - P_{ra}) { 1 - (1 - P_{sr})P_r (1 - P_m) } { P_{sr} + (1 - P_{sr}) (1 - P_{sr}P_r + P_{sr}P_rP_m) } +$</td>
</tr>
<tr>
<td>Constant</td>
<td>Constants expressed by parameters (switching AND-repair)</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td><strong>A</strong></td>
<td>[ P_{sr}(1 - P_{sr})^2 P_{r}^2 (P_{m} - P_{m})^3 ] + [ 2 P_{sr} P_{r} (1 - P_{sr}) P_{r}^2 (P_{m} - P_{ra})^2 { 1 - (1 - P_{sr}) P_{r}(1 - P_{ra}) } + [ P_{r}^2 P_{sr} P_{r}(P_{m} - P_{ra}){ 1 - (1 - P_{sr}) P_{r}(1 - P_{ra}) }^2 ] + [ P_{r}^2 (1 - P_{sr} P_{r}){ 1 - (1 - P_{sr}) P_{r} }^2 ]</td>
</tr>
<tr>
<td><strong>B</strong></td>
<td>[ (1 - P_{sr}) P_{r}^2 (P_{m} - P_{ra})^2 { 3 P_{sr} P_{r}(1 - P_{m})(1 - P_{sr} - P_{sr} - 1) - [ 4 P_{sr} P_{r} P_{r}(P_{m} - P_{ra}){ 1 - (1 - P_{sr}) P_{r}(1 - P_{ra}) } + [ 1 - (1 - P_{sr}) P_{r}(1 - P_{ra}) }^2 - P_{r}(2 + P_{r})(1 - P_{sr} P_{r}){ 1 - (1 - P_{sr}) P_{r} }^2 ]</td>
</tr>
<tr>
<td><strong>C</strong></td>
<td>[ P_{sr} P_{r}{ 3(1 - P_{m})^2 (P_{m} - P_{ra}) - 1 } ] + [ P_{sr} P_{r}{ 2(1 - P_{m})(P_{m} - P_{ra}){ -3 P_{r}(1 - P_{m}) + 1 } + 2 P_{r} - 1 } + [ P_{sr} P_{r}{ (P_{m} - P_{ra}){ 3 P_{r}^2 (1 - P_{m})^2 - 1 } - P_{r}^2 + 1 } + 2 P_{r}(P_{m} - P_{ra})(P_{m} - P_{r} + 1) + P_{r}(P_{r} - 2) + [ 2 P_{sr} P_{r}(P_{m} - P_{m}){ 1 - (1 - P_{sr}) P_{r}(1 - P_{ra}) } + (1 - P_{sr}) P_{r}{ 1 - (1 - P_{sr}) P_{r} }^2 ]</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>[ P_{r}(1 - P_{m}){ 1 - (1 - P_{sr}) P_{r}(1 - P_{m}) }{ P_{sr} + (1 - P_{sr}) (1 - P_{sr} P_{r} + P_{sr} P_{m}) } + [ (1 - P_{sr}) P_{r}(1 - P_{m}) ]</td>
</tr>
</tbody>
</table>
## Index

**A**  
Abnormal nodes eradication, 45, 92, 127  
Action error rate, 66  
Active phase, 103  
Active region, 45  
Actuator, v  
Adaptive defense, 38  
Agent, 11, 13, 20  
  selfish agent, 31  
Agent-based simulation, 9, 40, 46  
All-C, 64  
All-D, 64, 78  
Always repair, 56  
AND repair, 87, 99  
Asymmetric interaction, 74  
Asymmetry inherited from the existence property, 42  
Asymmetry of existence and non-existence, 2  
Autonomic computing, 8  
Availability, 25, 30  
Available resource, 64

**D**  
Damage rate, 64  
DDOS attacks, 23  
Defectors, 64  
De Morgan's theorem, 88  
Domany-Kinzel model, 42, 89  
Double edged sword, 23, 62  
Dual causality, 4  
Duality, 95  
Dynamic failure rate, 70

**E**  
Eight neighbors, 77  
Epidemic model, 129  
  SI model, 129  
  SIR model, 129  
  SIS model, 129  
Evolutionarily stable strategy (ESS), 7, 9, 73  
Extended phase diagram, 112, 113

**F**  
Failure rate, 26, 27, 63  
Fault probability, 25  
First order transition, 38  
Free-ride, 64  
Free-rider, 25, 60  
Frozen phase, 103  
Frozen region, 45

**G**  
Generalized SPD, 52  
Grid computing, 8

**H**  
Hamilton’s rule, 31  
Hawk-Dove, 25  
Heterogeneous agents, 22
<table>
<thead>
<tr>
<th>Page</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>156</td>
<td></td>
</tr>
<tr>
<td>156</td>
<td></td>
</tr>
</tbody>
</table>

**I**
- Immune system, 38, 81
- Immunity-based systems, 3
- Infection, 101
- Initial configuration, 56
- Iterated prisoner’s dilemma (IPD), 25

**K**
- $k$-D strategy, 53, 78
- Kin selection, 31
- Kolmogorov equation, 26, 27

**L**
- Lattice space, 54
- Local minima, 67

**M**
- Markov model, 26, 27
- Master equation, 130
- Maximum resource, 64
- Mean field approximation, 44, 91, 101
- Mean field assumption, 44
- Mechanism design, 21
- distributed algorithmic mechanism design, 21
- Mechanism of the strategy copy in repair, 61
- Membrane formation, 78
- Mixed repair, 110
- Moore neighborhood, 54, 63, 77
- Multiagent simulation, 46
- Mutually supporting collectives, 33
- Mutual repair, 99

**N**
- Nash equilibrium, 9, 21, 25, 86
- Neighborhood radius, 77
- Network cleaning problem, 13, 23
- Node, 11, 20
- Nowak-May’s simulations, 52
- Numeric analysis, 40

**O**
- Ordered parameter, 38
- OR repair, 87, 99

**P**
- Parasitic computing, 7
- Payoff matrix, 52
- Periodic boundary condition, 63
- Phase transition, 37, 38
- Prisoner’s dilemma, 11, 74
- Probabilistic cellular automata, 39
- Probabilistic cellular automaton, 42, 88
- Probability
  - probabilistic measure, 25
- Public goods game, 25, 60, 81

**R**
- Random attacks, 38
- Rate, 25
- Recovery oriented computing, 8
- Reliability, 25
- Repair rate, 26, 27, 41, 63, 87
- Repair success rate, 27
- Repair success rate by abnormal node, 42
- Repair success rate by normal node, 41
- Replicator dynamics, 63
- Resource
  - consumed resources, 64

**S**
- Scale-free network, 38
- Second-order cellular automaton, 76
- Second order transition, 38
- Selective attacks, 38
- Self-dual, 89, 95
- Self-involvement, 1
- Selfish agent, 31
- Self-recognition computers, 8
- Self-recognition model, v
- Self repair, 99
- Self-star property, 8
- Sensor, v
- Simultaneous repair, 87, 103
- Single repair, 103
- SIS model, 129
- Snow Drift, 82
- Spam mail, 60, 74
- Spatial generosity, 53, 81
- Spatial prisoner’s dilemma (SPD), 12, 25
- Spatial strategy, 52, 77
- Spatiotemporal strategies, 75
- Spontaneous recoveries, 38
- Steady state, 42, 44
- Steady-state probability, 28
- Strategic repair, 56
- Strategy code, 54
- Strategy update cycle, 55, 64
Structure function, 13
Switching repair, 110
Systemic payoff, 30, 61, 64

T
Temporal strategy, 75, 77
The mechanism of the extended payoff, 61
Tit-for-tat (TFT), 12, 81
Totalistic spatial strategy, 77

Transient phase diagram, 115
Transient state, 115

U
Unit in biological system, 4

W
Worms, 38