รายงานการวิจัยฉบับสมบูรณ์

ชื่อโครงการวิจัย (ภาษาไทย)

การเปลี่ยนแปลงเชิงพลวัตรของก้อนเนื้องอกกับการแข่งขันด้วยระบบภูมิคุ้มกัน

และแบบจำลองเซลลูลาร์ออโตเมตอนที่แสดงการเจริญของเนื้องอก

(ภาษาอังกฤษ) **Dynamics of the tumor with immune system competition using a cellular automaton model of tumor growth.**

โดย ผู้ช่วยศาสตราจารย์อังคณา บุญดิเรก

Assistant Professor Ankana Boondirek

หัวหน้าโครงการวิจัย

ทุนอุดหนุนการวิจัย ประเภทเงินรายได้

ประจำปีงบประมาณ พ.ศ. 2556 ตามมติคณะรัฐมนตรี

คำนำ

เนื่องด้วยผู้วิจัยได้รับการสนับสนุนทุนวิจัยเพื่อพัฒนางานวิจัยในหัวข้อเรื่อง “การเปลี่ยนแปลงเชิงพลวัตรของก้อนเนื้องอกกับการแข่งขันด้วยระบบภูมิคุ้มกัน และแบบจำลองเซลลูลาร์ออโตเมตอนที่แสดงการเจริญของเนื้องอก” ได้รับทุนอุดหนุนการวิจัย งบประมาณรายได้(เงินอุดหนุนจากรัฐบาล) ประจำปีงบประมาณ พ.ศ. ๒๕๕๖ ซึ่งเป็นงานวิจัยเชิงคำนวณเชิงชีววิทยา เพื่ออธิบายการเจริญของเนื้องอกกับการทำปฏิกิริยกับภูมิคุ้มกันของร่างกายในแง่มุมที่ปรากฏ มีความคาดหวังไว้ว่างานวิจัยนี้จะเกิดผลประโยชน์หลายด้าน เช่น ตีพิมพ์เผยแพร่ผลงานวิจัย และการขยายงานวิจัยเพื่อเชื่อมโยงให้เกิดประโยชนในด้านหนึ่งด้านใด อย่างแท้จริง

กรกฎาคม 2558

อังคณา บุญดิเรก

ส่วนที่หนึ่ง ส่วนนำ

1.1 Title Research

(ภาษาไทย) **การเปลี่ยนแปลงเชิงพลวัตรของก้อนเนื้องอกกับการแข่งขันด้วยระบบภูมิคุ้มกัน และแบบจำลองเซลลูลาร์ออโตเมตอนที่แสดงการเจริญของเนื้องอก**

(ภาษาอังกฤษ) **Dynamics of the tumor with immune system competition using a cellular automaton model of tumor growth.**

1.2 Acknowledgement

ขอขอบพระคุณมหาวิทยาลัยบูรพาที่ให้การสนับสนุนทุนวิจัย เพื่อทำงานวิจัย ซึ่งเป็นทุนอุดหนุน ประจำปีงบประมาณ พ.ศ. 2556 ตามมติคณะรัฐมนตรี

ส่วนที่สอง ส่วนเนื้อความ

2.1 บทนำ (Introduction)

As known, cancer is leading cause of human death in many past decade. In tumor biology, cancer is uncontrolled growth of cells and also has the ability to invade and damage normal tissues. In microscopic mechanisms, the immune system which attack tumor cells is complex with principal role to suppress the cancer or tumor growth in both *in vivo* and *in vitro*, see detailed in [1-9]. The important roles of immune system to cancer cell start with antigen presenting cells (APCs) to recognize foreign antigens and present to T cells to induce the immune cells to recognize and memorized cancer cells, as called immune surveillances. However, after the immune surveillances recognized the cancer cells and form to be the cancer –complexes cells, they can evade the immune system to be malignant tumor [5, 6]. The role of immune response after recognition process, will be the duty of cytotoxic immune cells, such as CD8+ or T cells, NK cells these cells may called effector cells, to kill the tumor cells to be dead tumor cell or necrosis cell [5-9]. The apoptosis of dead tumor cells with degradation process from the tissues [5,6,9], is caused by the effective immune response. However, if the dead tumor cells keep dormant or necrosis is caused by ineffective immune response, as immunosuppression to induce apoptosis and decay process of the dead tumor cells [5-8]. The explanations of the effective and ineffective immune response to show the critical role for immune system to control the tumor, we can see the detail in [5, 6].

2.2 การสำรวจแนวความคิดและการวิจัยที่เกี่ยวข้อง (Survey of Related Literature)

As known, that cancer growth with immune response in subcellular level is complex biological processes and poorly understood, which challenges the mathematicians and computational scientist to explain the cancer growth, as seen the various mathematical models of cancer growth with immune interaction, as reviewed in [10,11]. A number of computational researches proposed the mathematical model to explain the tumor growth with immune interactions over the past decade, such as [12-18]. The cellular automata model (CA) is a mathematical model which suitable for study the emergent behavior of microscopic interaction processes including the tumor –immune interactions such as [12-17]. In 1993, Qi and coworker [12], they proposed the kinetic model to explain the cell dynamics for tumor growth with immune response to tumor and then employed the stochastic cellular automata model (SCA) on two dimensional square lattice. They studied the simulation results by the growth curves with the difference set of parameter setting. The growth curve give rise the Gompertz – like curves, which is well known for predict the avascular tumor growth in experimental results [20,21]. In 2006, Boondirek and coworker presented the SCA

of cancer growth with immune response on two dimensional square lattices by adding the escape mechanism for cancer cells from immune surveillance which found in experimental reports, as see the detail in [5,13,19]. Boondirek and Triampo[14] studied the kinetic model which proposed by [13] to extend the SCA model on cubic lattice with more million cells for their simulation results with compared with experimental data, and Boondirek and coworker [15] extended study to give rise the simulation results for the sensitivity analysis for some parameters in their model to explore the efficacy of parameter and compared the simulation results with the experimental results.

2.3 วิธีการวิจัย (Procedure)

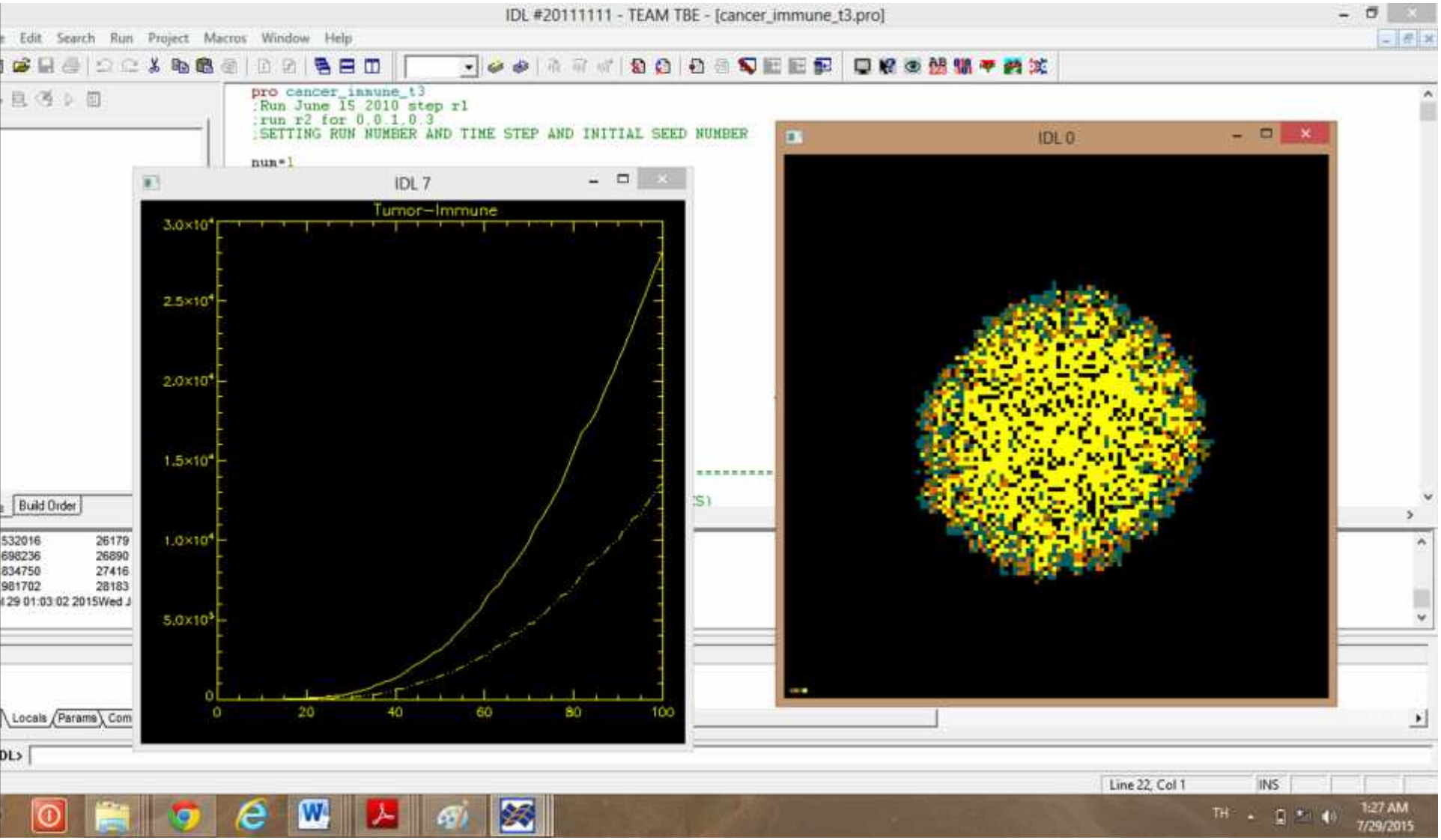
1. ค้นคว้า ผลการทดลองทางคลินิกของการเจริญของเนื้องอก จากผลการทดลองที่ตีพิมพ์
2. สร้างโดยพัฒนาแบบจำลองทางคณิตศาสตร์เพื่ออธิบายการเจริญของเซลเนื้องอกกับการยังยั้งการเจริญจากผลตอบสนองของภูมิคุ้มกันในระดับเซล
3. สร้างตัวแบบเซลลูลาร์ออโตเมตอนของการเจริญของเนื้องอกกับการตอบสนองด้วยภูมิคุ้มกัน จากการสร้างกฎของออโตเมตา
4. เขียนโปรแกรมคอมพิวเตอร์ ตามตัวแบบเซลลูลาร์ออโตเมตอนของการเจริญของเนื้องอกกับการตอบสนองด้วยภูมิคุ้มกัน
5. ทดสอบความถูกต้องของโปรแกรมที่เขียน
6. เปรียบเทียบข้อมูลการทดลองทางคลินิกกับข้อมูลที่ได้จากการทดลองเชิงคอมพิวเตอร์
7. เขียน manuscriptและเสนอต่อสำนักพิมพ์เพื่อตีเผยแพร่ผลงานในระดับนานาชาติ

2.4 ผลการวิจัย (Result)

2.4.1 ผลการวิจัยจากการรันโปรแกรม cancer_immune_t3.pro แนบไว้ที่ภาคผนวก

มีผลการรัน สองส่วน คือ ส่วนรันผลเป็นกราฟฟิก และเป็นข้อมูล

ส่วนที่หนึ่ง เป็นผลจากการรันเพื่อแสดงผลเป็นกราฟฟิก


IDL #20111111 - TEAM TBE - [cancer_immune_t3.pro]
Edit Search Run Project Macros Window Help
pro cancer_immune_t3
;Run June 15 2010 step r1
;run r2 for 0.0,1.0,3
;SETTING RUN NUMBER AND TIME STEP AND INITIAL SEED NUMBER
num=1
IDL 7
Tumor-Immune
3.0×10⁴
2.5×10⁴
2.0×10⁴
1.5×10⁴
1.0×10⁴
5.0×10³
0
0
20
40
60
80
100
IDL 0
Build Order
532016 26179
698236 26890
834750 27416
981702 28183
Locals Params Com
Line 22, Col 1
INS
TH
1:27 AM
7/29/2015

∞

ส่วนที่สอง คือ ผลข้อมูล cancer_immune_t3.txt

	N(t)	T(t)	C(t)	D(t)
1	10.0000	6.00000	4.00000	0.000000
2	12.0000	4.00000	5.00000	0.000000
3	13.0000	4.00000	2.00000	0.000000
4	14.0000	2.00000	4.00000	1.00000
5	13.0000	2.00000	2.00000	2.00000
6	11.0000	4.00000	0.000000	3.00000
7	12.0000	4.00000	2.00000	5.00000
8	13.0000	4.00000	2.00000	5.00000
9	16.0000	9.00000	0.000000	6.00000
10	20.0000	8.00000	5.00000	6.00000
11	23.0000	9.00000	4.00000	6.00000
12	26.0000	10.0000	5.00000	6.00000
13	35.0000	19.0000	4.00000	6.00000
14	41.0000	20.0000	9.00000	6.00000
15	52.0000	25.0000	10.0000	8.00000
16	63.0000	35.0000	10.0000	10.0000
17	85.0000	50.0000	13.0000	12.0000
18	113.000	61.0000	23.0000	14.0000
19	141.000	78.0000	27.0000	16.0000
20	174.000	99.0000	35.0000	19.0000
21	215.000	120.000	40.0000	26.0000
22	255.000	122.000	65.0000	34.0000
23	294.000	137.000	66.0000	39.0000
24	337.000	162.000	73.0000	46.0000
25	406.000	200.000	77.0000	57.0000
26	473.000	229.000	97.0000	72.0000
27	558.000	257.000	116.000	91.0000
28	641.000	292.000	119.000	110.000
29	702.000	316.000	156.000	131.000
30	828.000	395.000	143.000	152.000
31	962.000	452.000	191.000	181.000
32	1124.00	546.000	198.000	204.000
33	1319.00	633.000	246.000	242.000
34	1509.00	704.000	315.000	267.000
35	1711.00	789.000	334.000	313.000
36	1965.00	921.000	351.000	369.000
37	2228.00	1029.00	424.000	435.000
38	2538.00	1142.00	484.000	508.000

39	2854.00	1259.00	546.000	584.000
40	3185.00	1342.00	658.000	665.000
41	3506.00	1484.00	662.000	770.000
42	3912.00	1692.00	696.000	894.000
43	4378.00	1886.00	808.000	1018.00
44	4897.00	2081.00	904.000	1160.00
45	5403.00	2261.00	994.000	1305.00
46	5948.00	2382.00	1116.00	1478.00
47	6504.00	2580.00	1171.00	1673.00
48	7082.00	2786.00	1294.00	1901.00
49	7744.00	3013.00	1344.00	2125.00
50	8344.00	3168.00	1480.00	2396.00
51	9031.00	3425.00	1540.00	2662.00
52	9747.00	3599.00	1695.00	2918.00
53	10522.0	3940.00	1714.00	3228.00
54	11332.0	4233.00	1926.00	3551.00
55	12317.0	4530.00	2073.00	3881.00
56	13264.0	4868.00	2198.00	4241.00
57	14220.0	5034.00	2401.00	4610.00
58	15182.0	5361.00	2474.00	5047.00
59	16237.0	5678.00	2649.00	5490.00
60	17371.0	6131.00	2757.00	5982.00
61	18698.0	6595.00	2925.00	6482.00
62	19986.0	6922.00	3197.00	7042.00
63	21297.0	7176.00	3435.00	7625.00
64	22670.0	7582.00	3511.00	8197.00
65	23997.0	8074.00	3610.00	8878.00
66	25458.0	8321.00	3993.00	9589.00
67	26913.0	8640.00	4153.00	10288.0
68	28427.0	9050.00	4241.00	11108.0
69	29962.0	9467.00	4453.00	11899.0
70	31574.0	9856.00	4748.00	12711.0
71	33412.0	10579.0	4794.00	13559.0
72	35336.0	11077.0	5109.00	14453.0
73	37228.0	11520.0	5351.00	15392.0
74	39181.0	12003.0	5623.00	16346.0
75	41225.0	12409.0	5965.00	17382.0
76	43297.0	12957.0	6124.00	18438.0
77	45430.0	13433.0	6461.00	19566.0
78	47764.0	14081.0	6591.00	20702.0
79	49988.0	14794.0	6781.00	21949.0

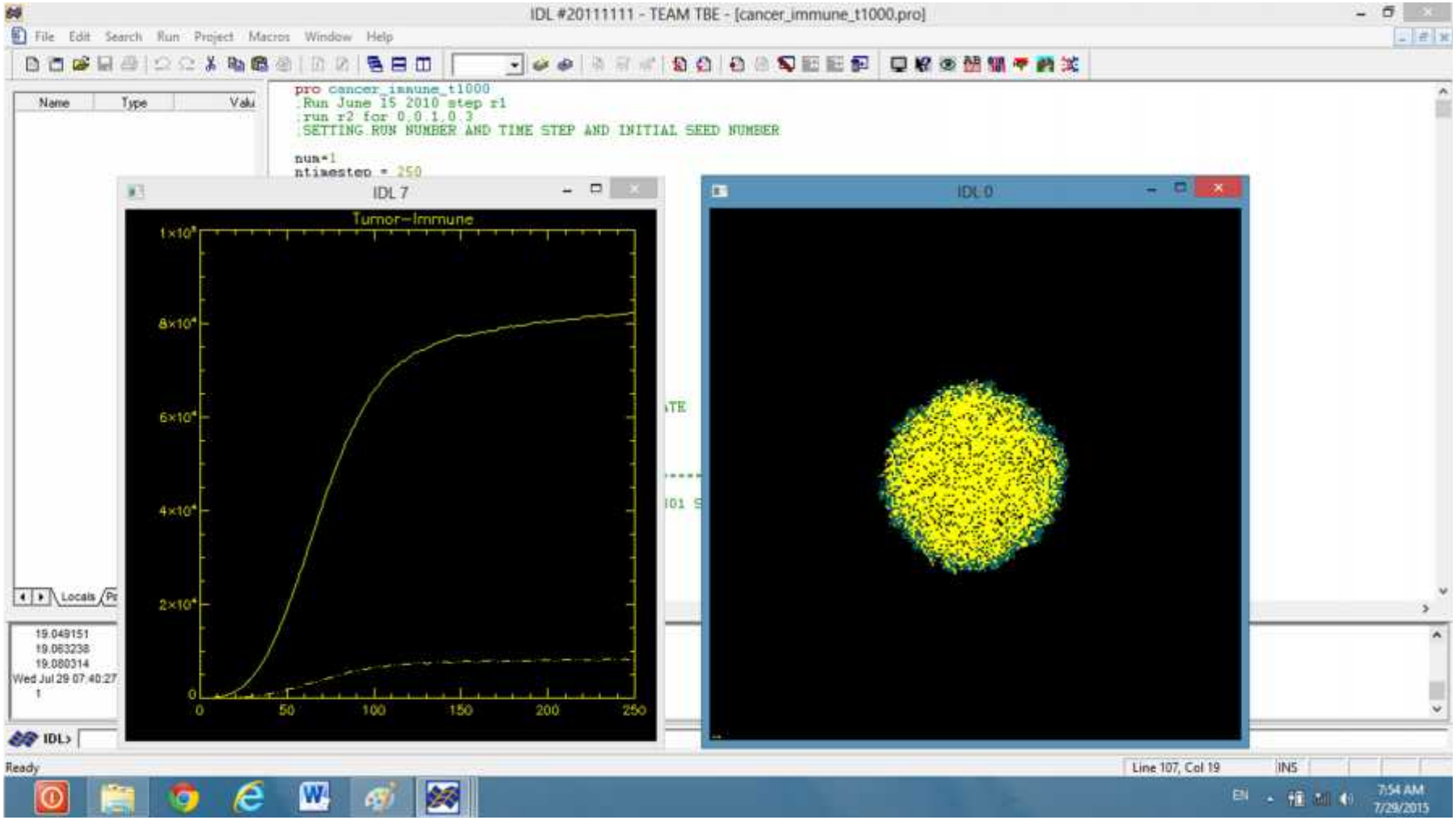
80	52499.0	15446.0	7171.00	23232.0
81	55072.0	16149.0	7505.00	24524.0
82	57794.0	16812.0	7852.00	25821.0
83	60384.0	17105.0	8404.00	27263.0
84	63089.0	17613.0	8527.00	28783.0
85	65760.0	18059.0	8781.00	30403.0
86	68467.0	18763.0	8946.00	32076.0
87	71320.0	19443.0	9248.00	33767.0
88	74205.0	19974.0	9642.00	35581.0
89	77242.0	20550.0	9956.00	37448.0
90	80379.0	21374.0	9984.00	39267.0
91	83421.0	21809.0	10544.0	41194.0
92	86847.0	22590.0	10786.0	43182.0
93	90263.0	23179.0	11234.0	45266.0
94	93662.0	23928.0	11457.0	47372.0
95	97255.0	24750.0	11672.0	49585.0
96	100764.	25451.0	12127.0	51762.0
97	104608.	26179.0	12535.0	54096.0
98	108561.	26890.0	12922.0	56422.0
99	112306.	27416.0	13370.0	58901.0
100	116114.	28183.0	13564.0	61338.0

From 01:03:02 2015 to 01:04:20 2015

2.4.2 ผลการวิจัยจากการรันโปรแกรม cancer_immune_t1000.pro แนบไว้ที่ภาคผนวก

มีผลการรัน สองส่วน คือ ส่วนรันผลเป็นกราฟฟิก และเป็นข้อมูล

ส่วนที่หนึ่ง เป็นผลจากการรันเพื่อแสดงผลเป็นกราฟฟิก


IDL #20111111 - TEAM TBE - [cancer_immune_t1000.pro]
File Edit Search Run Project Macros Window Help
Name Type Valu
pro cancer_immune_t1000
;Run June 15 2010 step r1
;run r2 for 0.0,1.0,3
;SETTING RUN NUMBER AND TIME STEP AND INITIAL SEED NUMBER
num=1
ntimestep = 250
IDL 7
Tumor-Immune
1x10^5
8x10^4
6x10^4
4x10^4
2x10^4
0
0
50
100
150
200
250
IDL 0
Locals
19.049151
19.083238
19.080314
Wed Jul 29 07:40:27
1
IDL>
Ready
Line 107, Col 19
INS
EN
7:54 AM
7/29/2015

2.4.4 ต้นฉบับเพื่อขอส่งเสนอตีพิมพ์(รอปรับปรุงจากการส่งเสนอตีพิมพ์)

July, 29, 2015

A stochastic cellular automata model of avascular tumor growth with immune surveillance and response against tumor: cubic lattice

Ankana Boondirek, I Ming Tang, Kanint Teerapabolarn, Apisit Pakapongpun, and Wannapong Triampo*

Keywords: Stochastic model, *in silico* model of tumor growth, cellular automata model in 3D cubic lattice, and tumor growth with immune response

Abstract

The kinetic model of an avascular tumor growth with tumor-host immune interaction with six parameters is presented. The model takes to account the immune surveillance to recognize tumor cell, and immune progression for cytotoxic activity and apoptosis of dead tumor cells including the immune suppression to escape of tumor cells by immune surveillance, and the dormancy of dead tumor cells by ineffective immune responses. A stochastic cellular automata model on a 3D cubic lattice is hybrid to implement the kinetic model. The simulation results, such as the growth curves will be discuss.

1. Introduction

As known, cancer is leading cause of human death in many past decade. In tumor biology, cancer is uncontrolled growth of cells and also has the ability to invade and damage normal tissues. In microscopic mechanisms, the immune system which attack tumor cells is complex with principal role to suppress the cancer or tumor growth in both *in vivo* and *in vitro*, see detailed in [1-9]. The important roles of immune system to cancer cell start with antigen presenting cells (APCs) to recognize foreign antigens and present to T cells to induce the immune cells to recognize and memorized cancer cells, as called immune surveillances. However, after the immune surveillances recognized the cancer cells and form to be the cancer –complexes cells, they can evade the immune system to be malignant tumor [5, 6]. The role of immune response after recognition process, will be the duty of cytotoxic immune cells, such as CD8+ or T cells, NK cells these cells may called effector cells, to kill the tumor cells to be dead tumor cell or necrosis cell [5-9]. The apoptosis of dead tumor cells with degradation process from the tissues [5,6,9], is caused by the effective immune response. However, if the dead tumor cells keep dormant or necrosis is caused by ineffective immune response, as immunosuppression to induce apoptosis and decay process of the dead tumor cells [5-8]. The explanations of the effective and ineffective immune response to show

the critical role for immune system to control the tumor, we can see the detail in [5, 6].

As known, that cancer growth with immune response in subcellular level is complex biological processes and poorly understood, which challenges the mathematicians and computational scientist to explain the cancer growth, as seen the various mathematical models of cancer growth with immune interaction, as reviewed in [10,11]. A number of computational researches proposed the mathematical model to explain the tumor growth with immune interactions over the past decade, such as [12-18]. The cellular automata model (CA) is a mathematical model which suitable for study the emergent behavior of microscopic interaction processes including the tumor –immune interactions such as [12-17]. In 1993, Qi and coworker [12], they proposed the kinetic model to explain the cell dynamics for tumor growth with immune response to tumor and then employed the stochastic cellular automata model (SCA) on two dimensional square lattice. They studied the simulation results by the growth curves with the difference set of parameter setting. The growth curve give rise the Gompertz – like curves, which is well known for predict the avascular tumor growth in experimental results [20,21]. In 2006, Boondirek and coworker presented the SCA of cancer growth with immune response on two dimensional square lattices by adding the escape mechanism for cancer cells from immune surveillance which found in experimental reports, as see the detail in [5,13,19]. Boondirek and Triampo[14] studied the kinetic model which proposed by [13] to extend the SCA model on cubic lattice with more million cells for their simulation results with compared with experimental data, and Boondirek and coworker [15] extended study to give rise the simulation results for the sensitivity analysis for some parameters in their model to explore the efficacy of parameter and compared the simulational results with the experimental results.

The aim of this research is to hybrid the SCA in 3D cubic lattice with the kinetic model, presenting by the realistic emergent of tumor growth with immune system interaction from the studies of the clinical, experimental data, and mathematical models in [1-9,12-19] and then investigate the simulation results.

This paper is organized as follows: The model design rationale and the method of the SCA model are presented in section II. In section III, we present the simulation results for tumor progression and suppression. The conclusion and discussion will be given in section IV.

2. **Outline of the SCA model**

2.1 A Kinetic Model for Avascular Tumor Growth with Immune Interactions

We propose the kinetic model to present the avascular tumor growth with immune surveillance, effective and ineffective immune response.

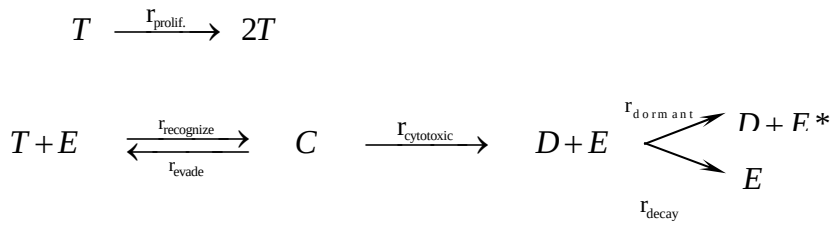


Fig. 1 The kinetic model of cancer cells with immune surveillance and suppression. T, C, D denotes the number of tumor, tumor-immune complexes, dead tumor cells, and dormant cells, respectively. And E, E* denotes the effective and ineffective immune cells,

The first reaction, representing the proliferation of tumor cells by mitosis process, the proliferative rate of *in vivo* tumor is defined by, $r_{\text{prolif.}} = r_{\text{p}}(1-\frac{T}{T_{\max}})$ which is a function of r_{p}, and the number of tumor cells, T. Where $T_{\max}$ denotes as the carrying capacity for the number of proliferating tumor cells [12-15].

The second reaction, indicating the immune surveillance to recognize to tumor cells and escape of tumor cells from interaction of immune recognition. The recognize rate of immune surveillance is defined by, $r_{\text{recognize}} = r_{\text{r}}(1-\frac{C}{C_{\max}})$. Where, r_{r} is the recognize parameter, and C is the number of tumor – immune complexes cells , C which indicate the number of tumor-immune complexes cells, and $C_{\max}$ is the carrying capacity of the effector cells for recognition to tumor cells. The tumor cells can evade from tumor – immune complexes cells with the evading rate, r_{evade}. The tumor – immune complexes cells can be change to be dead tumor cell by cytotoxic activity by immune cells with cytotoxic rate, $\text{r}_{\text{cytotoxic}}$. By the last reaction, it indicates the both effective and ineffective immune responses, by the dormant rate, r_{dormant} which is the rate of dead tumor cell interacting with the ineffective immune activity; the immune system cannot induce the dead tumor cell to programmed cell death or apoptosis and keeping dormant. But in the case of effective immune response to dead tumor cells, the degradation of dead tumor cells or programmed cell death which make the dead cells loss from the tumor cluster with the decay rate, r_{decay}. Where, E and E* represents the effective immune cells and ineffective immune cells, respectively.

Fig. 2 This figure shows the initial condition of the seven proliferative cells in the middle of the cubic lattice in the rule for CA. This CA 's rule uses von – Neumann neighborhood for dynamical change, the nearest neighboring cells of the middle one(unseen cell) are located around the middle one, which can be seen.

2.2 The method of the SCA model

2.2.1 The model formulation

The biological assumptions, which we took into account when developing the kinetic model are based on both clinical and experimental data between tumor growth with immune system from [1-9] and the mathematical models as in [10-19]. The assumptions are as followed:

1. Tumor is uncontrolled growth of cells, they can divide by mitosis, only they have enough food and space, and they can divide to two daughter tumor cells with proliferative rate, $r_{prolif.}$
2. The immune system will have the three mechanism steps to tumor cells, recognition mechanism to the proliferative tumor cells by forming them to be tumor – complexes cells, cytotoxic mechanism by changing the tumor – complexes cells to be dead tumor cells, and apoptosis mechanism to degrade the dead tumor cells to loss them with the program cell death, respectively.
3. By the recognition mechanism, the proliferative tumor cells are recognized by immune surveillance with recognize rate, $r_{recognize}$ and then form to tumor-immune complexes cells.
4. The tumor cells escape from tumor – immune complexes cells by the ineffectiveness of immune surveillance to be proliferative tumor cells with evade rate, r_{evade}.
5. By the cytotoxic mechanism, the effective immune response within tumor – immune complexes cells changes the tumor to be the dead tumor cells and effective immune cells with the cytotoxic rate, $r_{cytotoxic}$.
6. The dormancy mechanism, caused by the ineffective immune response to eliminate the dead tumor cell to apoptosis, the dead tumor kept dormant, we

can say that the immune suppression to bring the dead tumor cell in programmed cell death with dormant rate, $r_{dormant}$.

7. The degradation mechanism, caused by the effective immune cells eliminate the dead tumor cell to vanish from the tissues with the decay rate, r_{decay}.

2.2.2 The Algorithm of SCA model

The tissue model is rely on 3D cubic lattice with size, $L \times L \times L$. The details of algorithm for a simulation can be described as the following steps.

Step I. The initial time step: $t = 0$

The initial condition in CA model is the seven proliferative tumor cells, which are located in the center of the cubic lattice as seen in Fig. 2.

The parameter rate are set, which are $r_p, r_r, r_{evade}, r_{cytotoxic}, r_{dormant}, r_{decay}$ C_{max}, and T_{max}. The setting parameters have to satisfy the conditions, that each rate have the range between 0 and 1, and $r_p + r_r \le 1, r_{evade} + r_{cytotoxic} \le 1, r_{dormant} + r_{decay} \le 1$.

The initial configuration for tumor and complexes cells is $T(0) = 7$ and $C(0) = 0$.

Step II. The next time step: $t = t + 1$, starting at $t = 1$

We will describe the details of the automata-based model. The synchronously updating in the discrete space and time with von Neumann neighborhood in SCA model is set up. That is the state of each site on the cubic lattice can update at most once per time step. The dynamics of cells is rely on the random number chosen, space conditions and its rate indicates by the schematics diagram, in Fig.3.

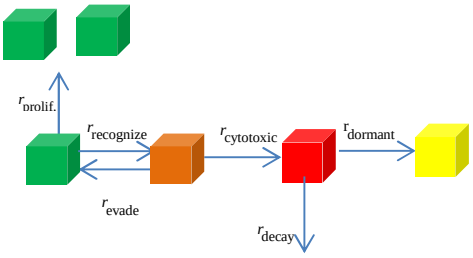


Fig. 3 The diagram is representing the tumor cell dynamics with immune competition from the kinetic model. (= proliferative tumor cell, = complexes cell, = dead tumor cell, and = dormant cell)

For each proliferative tumor cell at time *t*, after we picked a random number, and calculated the proliferative rate, the recognize rate, and considered the condition for space to make the decisions with three possible events:

(i) Change to be the complexes cells, *C* with the probability of the recognize rate, $r_{\text{recognize}}$ caused by immune surveillance. The recognize rate is calculated by $r_{\text{recognize}} = r_{\text{r}}(1 - \frac{C(t-1)}{C_{\max}})$, where $C(t-1)$ is the number of complexes cells in the former time step.

(ii) Duplicate to the two daughters' tumor cells, with the probability of the proliferative rate $r_{\text{prolif.}}(t)$, and there is at least one vacant nearest neighboring site of tumor cell. Otherwise, it will keep stay in the tumor state for the third case. The nearest neighboring of each tumor cell can be seen in Fig. 2. The proliferative rate is calculated by $r_{\text{prolif.}}(t) = r_{\text{p}}(1 - \frac{T(t-1)}{T_{\max}})$ where T($t-1$) is the number of tumor cells in the former time step.

(iii) Keep stay in the same state with the probability of $1 - (r_{\text{recognize}}(t) + r_{\text{prolif.}}(t))$ caused by the proliferative rate of specific tumor cell, recognize rate of immune surveillance, and lack of space for tumor proliferation.

For each complexes cell at time *t*, after we picked a random number, calculated the evade rate, and used the cytotoxic rate to make the decisions with three possible events:

(i) Change to be dead tumor cell with the probability of cytotoxic rate, $r_{\text{cytotoxic}}$ caused by the effective for immune surveillance.

(ii) Change to be proliferative cell with the probability of evade rate, r_{evade} caused by the escape of tumor cell.

(iii) Keep stay in the same state with the probability of $1 - (r_{\text{cytotoxic}} + r_{\text{evade}})$, caused by ineffective immune response to kill tumor cells or inability of evading the tumor cells from complexes cells.

For each dead tumor cell at time *t*, after we picked a random number, used the dormant rate, and decay rate to make the decisions with two possible events:

(i) Change to be vacant cell with the probability of decay rate, r_{decay} caused by the effective immune response to make the dead tumor cells degrade from tissues.

(ii) release the ineffective immune cells with the probability of the rate $r_{dormant}$ caused by ineffective immune response which cannot degrade the dead tumor cells form the tissues.

(iii) Keep stay in the same state with the probability of the rate 1 - ($r_{decay} + r_{dormant}$) .

Technically, all cells are randomly selected one by one with equal probability to decision for its dynamics. We progress iteratively in time step by $t = t + 1$, until we reach the desired number of time steps.

Step III. The step II was applied with self-organizing method until reach the desired number of time steps. We will have a simulated tumor growth.

3. Simulation Results

We have written a program source code to implemented by using the algorithms for SCA model. To investigate the simulation results, we measured the tumor growth curves and tumor growth patterns as the evolution in time is shown in Fig. 4 and Fig. 5, respectively.

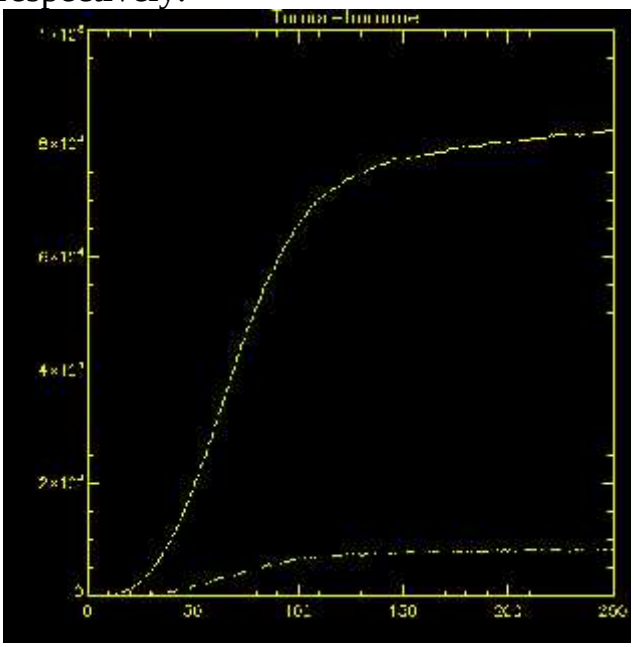


Fig. 4 Plots of time evolutions of the number of tumorous cells and number of tumor-immune complexes cells from a typical parameter setting is $r_p = 0.5, r_r = 0.05, r_{evade} = 0.1, r_{cytotoxic} = 0.4, r_{dormant} = 0.3, r_{decay} = 0.5$, and $T_{max} = C_{max} = 100000$.

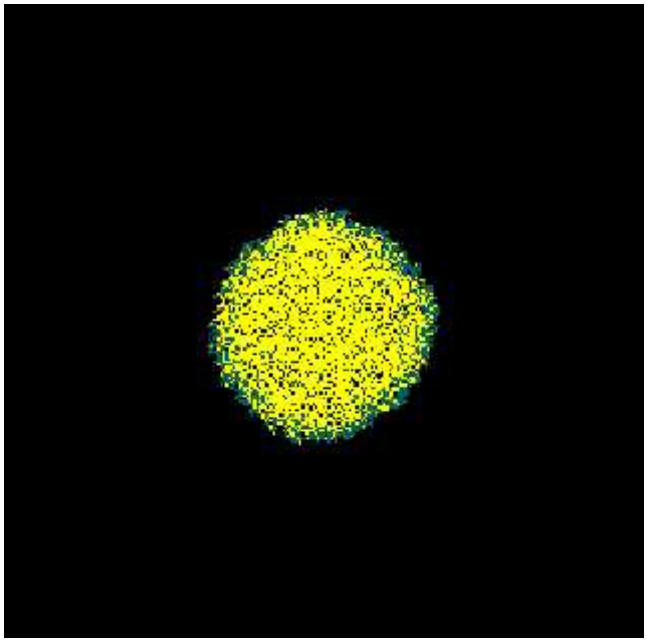


Fig. 5 The snapshot of simulated tumor shows the distribution of multicellular of cells generated by the SCA model hybrid the kinetic reaction is typically, shown. The simulated tumor with the four tumorous cell types, proliferative tumor cells, tumor-immune complexes cells, dead tumor cells, and dormant cells with the different color green, brown, red, yellow, respectively.
(= proliferative tumor cell, = complexes cell, = dead tumor cell, and = dormant cell)

The Fig.4 and 5. Show two results from a simulation setting. We use 250 time step and we can see qualitative of growth curve, which yields the Gompertz like curve. As known, the Gompertz growth curve is the most used curve to fit the experimental data for *in vivo* tumor growth [20-24]. Obviously, the outer most regions for tumor have contained the proliferating tumor cells as found by Bru et al. [25].

4. Conclusion and Discussion

In this research, we studied the tumor biology, in the effects tumor by immune system to propose the kinetic model for avascular tumor growth with tumor-host immune interaction, which are immune surveillance to recognize tumor cell, and immune progression for cytotoxic activity and apoptosis of dead tumor cells including the escape of tumor cells after they were activated by immune response and keep dormant of dead tumor cells. The tumor cells can evade after they were recognized by the immune system, and however, after the immune recognize the tumor cells and attack them to dead tumor cells, they can escape to program cell death cause by the immunosuppression. We created the cellular

automata model in cubic lattice to employ the kinetic model to simulate the tumor growth and found the Gompertz – like curves, which is the known mathematical function to explain the tumor growth. The computer simulation also shows the snapshot for simulated tumors in many time steps as seen in simulation sections. The growth curves shows as the function of the six parameters in the kinetic model, the simulation results can be show both the typical growth and decay tumor growth curves with the different parameter setting.

The Gompertz – like growth curves give rise by the simulated growth curves by a typical parameters setting as shown in Fig. 4.

Further we focus on comparing to the experimental tumor growth by calibrating the parameters to fit and relevant biological implications. The local sensitivity analysis of the parameters to explore the efficacy of each parameter to the simulation results can be calculated. Finally, the spatial distribution of proliferated tumor cell and complexity of simulated tumor can be measured.

5. References

[1] Abbas AK, Lichtman AH. Basic Immunology: Functions and Disorders of the Immune System. 3rd ed. Philadelphia, PA: Saunders Elsevier; 2011.

[2] Kasiske BL, Snyder JJ, Gilbertson DT, Wang C. Am J Transplant. 2004;4(6):905-913.

[3] Le Mire L, Hollowood K, Gray D, Bordea C, Wojnarowska F. Br J Dermatol. 2006;154(3):472-477.

[4] Kataki A, Scheid,P, Piet M, Marie B, Martinet N, Martinet Y and Vignaud J-M. Tumor Infiltrating Lymphocytes and Macrophages have a Potential Dual Role in Lung Cancer by Supporting both Host-Defense and Tumor Progression. Journal of Laboratory and Clinical Medicine. 2002; 140: 320-328.

[5] Hanahan D, Weinberg RA. Cell. 2011;144(5):646-674.

[6] Simone Mocellin, Francesco M. Marincola, and Howard A. Young. Journal of Leukocyte Biology Volume 78, November 2005

[7] Thomas F Gajewski, Hans Schreiber & Yang-Xin Fu. Innate and adaptive immune cells in the tumor microenvironment. NATURE IMMUNOLOGY VOLUME 14 NUMBER 10 OCTOBER 2013

[8] Bernhard Nieswandt, Michael Hafner, Bernd Echtenacher, and Daniela N. Mannel. Lysis of Tumor Cells by Natural Killer Cells in Mice Is Impeded by Platelets. CANCER RESEARCH 59, 1295–1300, March 15, 1999

[9] Parish, C. (2003). Cancer immunotherapy: the past, the present and the future. Immunol. Cell Biol., 81, 106–113

[10] Raluca Eftimie · Jonathan L. Bramson · David J.D. Earn. Bull Math Biol (2011) 73: 2–32. Interactions Between the Immune System and Cancer: A Brief Review of Non-spatial Mathematical Models.

[11] Preziosi, L. (2003). Cancer modeling and simulation. Chapman & Hall/CRC

[12] A. S. Qi, X. Zheng, C. Y. Du and B. S. An, J. Theor. Biol. 161, 1993.

[13] **Boondirek A** , Lenbury Y, Wong-ekkabut J, Triampo W, Tang I M, and Picha P. A Stochastic Model of Cancer Growth with Immune Response. Journal of the Korean Physical Society, Vol. 49, No. 4, October 2006; 1652-1666.

[14] Boondirek A, and W. Triampo W, Cancer Research: Computer Simulation of Tumor Growth with Immune Response, Naresuan University Journal, 17, 3 (2009) 196 – 200.

[15] Boondirek A, Teerapabolarn K., and Triampo W. A Stochastic Model of Tumor Growth With Immune Response: Three Dimensional Cubic Lattice. International Journal of Open Problems in Computer Science & Mat;Sep2011, Vol. 4 Issue 3, p29.

[16] A. Matzavinos, M.A. Chaplain, and V.A. Kuznetsov. Mathematical modeling of the spatio-temporal response of cytotoxic T-lymphocytes to a solid tumor. Math. Med. Biol. 21(1) pp.1-34 (2004)

[17] Lisette G. de Pillis, Ami E. Radunskaya, and Charles L. Wiseman, A Validated Mathematical Model of Cell-Mediated Immune Response to Tumor Growth. Cancer Res 2005; 65: (17). September 1, 2005

[18] Mallet, D.G. and de Pillis, L.G.(2006) A cellular automata model of tumor – immune system interactions. Journal of Theoretical Biology 239(3):pp. 334-350.

[19] Kuznetsov V.A. and Taylor M.A.(1994) : Nonlinear dynamics of immunogenic tumors: Parameter estimation and global bifurcation analysis. Bull. Math. Biol., Vol. 56, No. 2, pp.295-321.

[20] G.G. Steel, Growth Kinetics of Tumors, Clarendon Press, 1977.

[21] Norton, L. (1988). A Gompertzian model of human breast cancer growth. Cancer Res., 48, 7067–7071.

[23] R. Demicheli, G. Pratesi, and R. Foroni. The exponential-gompertzian growth model: data from six tumor cell lines in vitro and in vivo. estimate of the transition point from exponential to gompertzian growth and potential clinical applications. Tumori, 77:189–195, 1991.
[24] D.A. Cameron, A.A. Ritchie and W.R.Miller, Eur.J. Can. 37, 1545,2001
[25] A. Bru, S. Albertos, J.. Subiza and J.L.Garcia-Asenjo, Phys" Rev. Lett 92,23 238101, 2004

ส่วนที่สาม ส่วนอ้างอิงและผนวก

3.1 ส่วนอ้างอิง (References)

5. References

[1] Abbas AK, Lichtman AH. Basic Immunology: Functions and Disorders of the Immune System. 3rd ed. Philadelphia, PA: Saunders Elsevier; 2011.

[2] Kasiske BL, Snyder JJ, Gilbertson DT, Wang C. Am J Transplant. 2004;4(6):905-913.

[3] Le Mire L, Hollowood K, Gray D, Bordea C, Wojnarowska F. Br J Dermatol. 2006;154(3):472-477.

[4] Kataki A, Scheid,P, Piet M, Marie B, Martinet N, Martinet Y and Vignaud J-M. Tumor Infiltrating Lymphocytes and Macrophages have a Potential Dual Role in Lung Cancer by Supporting both Host-Defense and Tumor Progression. Journal of Laboratory and Clinical Medicine. 2002; 140: 320-328.

[5] Hanahan D, Weinberg RA. Cell. 2011;144(5):646-674.

[6] Simone Mocellin, Francesco M. Marincola, and Howard A. Young. Journal of Leukocyte Biology Volume 78, November 2005

[7] Thomas F Gajewski, Hans Schreiber & Yang-Xin Fu. Innate and adaptive immune cells in the tumor microenvironment. NATURE IMMUNOLOGY VOLUME 14 NUMBER 10 OCTOBER 2013

[8] Bernhard Nieswandt, Michael Hafner, Bernd Echtenacher, and Daniela N. Mannel. Lysis of Tumor Cells by Natural Killer Cells in Mice Is Impeded by Platelets. CANCER RESEARCH 59, 1295–1300, March 15, 1999

[9] Parish, C. (2003). Cancer immunotherapy: the past, the present and the future. Immunol. Cell Biol., 81, 106–113

[10] Raluca Eftimie · Jonathan L. Bramson · David J.D. Earn. Bull Math Biol (2011) 73: 2–32. Interactions Between the Immune System and Cancer: A Brief Review of Non-spatial Mathematical Models.

[11] Preziosi, L. (2003). Cancer modeling and simulation. Chapman & Hall/CRC

[12] A. S. Qi, X. Zheng, C. Y. Du and B. S. An, J. Theor. Biol. 161, 1993.

[13] **Boondirek A** , Lenbury Y, Wong-ekkabut J, Triampo W, Tang I M, and Picha P. A Stochastic Model of Cancer Growth with Immune Response. Journal of the Korean Physical Society, Vol. 49, No. 4, October 2006; 1652-1666.

[14] Boondirek A, and W. Triampo W, Cancer Research: Computer Simulation of Tumor Growth with Immune Response, Naresuan University Journal, 17, 3 (2009) 196 – 200.
[15] Boondirek A, Teerapabolarn K., and Triampo W. A Stochastic Model of Tumor Growth With Immune Response: Three Dimensional Cubic Lattice. International Journal of Open Problems in Computer Science & Mat;Sep2011, Vol. 4 Issue 3, p29.
[16] A. Matzavinos, M.A. Chaplain, and V.A. Kuznetsov. Mathematical modeling of the spatio-temporal response of cytotoxic T-lymphocytes to a solid tumor. Math. Med. Biol. 21(1) pp.1-34 (2004)
[17] Lisette G. de Pillis, Ami E. Radunskaya, and Charles L. Wiseman, A Validated Mathematical Model of Cell-Mediated Immune Response to Tumor Growth. Cancer Res 2005; 65: (17). September 1, 2005
[18] Mallet, D.G. and de Pillis, L.G.(2006) A cellular automata model of tumor – immune system interactions. Journal of Theoretical Biology 239(3):pp. 334-350.
[19] Kuznetsov V.A. and Taylor M.A.(1994) : Nonlinear dynamics of immunogenic tumors: Parameter estimation and global bifurcation analysis. Bull. Math. Biol., Vol. 56, No. 2, pp.295-321.
[20] G.G. Steel, Growth Kinetics of Tumors, Clarendon Press, 1977.
[21] Norton, L. (1988). A Gompertzian model of human breast cancer growth. Cancer Res., 48, 7067–7071.
[23] R. Demicheli, G. Pratesi, and R. Foroni. The exponential-gompertzian growth model: data from six tumor cell lines in vitro and in vivo. estimate of the transition point from exponential to gompertzian growth and potential clinical applications. Tumori, 77:189–195, 1991.
[24] D.A. Cameron, A.A. Ritchie and W.R.Miller, Eur.J. Can. 37, 1545,2001
[25] A. Bru, S. Albertos, J.. Subiza and J.L.Garcia-Asenjo, Phys" Rev. Lett 92,23 238101, 2004

3.2 ส่วนผนวก (Appendixes)

3.2.1 ตัวโปรแกรม ชื่อ cancer_immune_t3

```
pro cancer_immune_t3
;Run June 15 2010 step r1
;run r2 for 0,0.1,0.3
;SETTING RUN NUMBER AND TIME STEP AND INITIAL SEED NUMBER

num=1
ntimestep = 100

seed = 107L

t0 = systime()

  openw, 18, 'cancer_immune_t3.txt'
;SETTING PARAMETERS
akst=0.65  ;r_prolif
aktwo=0.35   ;r_binding
akndrev=0.1 ;r_detach
akrd= 0.6    ;r_lysis
akfour=0.4  ;r_decay
akfive=0.2   ;r dormant
iphi= 1000000ULL

;                LOOP  FOR RUN VARY EACH RATE              THE  BIGGEST LOOP

for iloo = 1, 1 do begin
akst  =  akst + 0.05*(iloo-1)
;               step*0.1
;=====================================================================

;SETTING THE LATTICE SIZE (MAX 301 X 301 X 301 SITES)

iLx=121ULL
iLy=iLx
```

```
icom= iLx*iLx*iLx
print, '    THE TOTAL SITES = ',icom


;SETTING SIZE OF LATTICE SHOWING

device, retain=2, decomposed=0
loadct,4

ixy_shw=500   ;SIZE OF SHOWING LATTICE
window, 0, xsize = ixy_shw, ysize = ixy_shw


;SHOW CUT X-PLANE Y-PLANE OR Z-PLANE
ishw = 1;(shw =1  cutx);ishw = 2;(shw cuty);ishw = 3;(shw cutz)
nx = 500
ny = 500

;SETTING CELLS TYPE VARIABLES

normal=0   ;BLACK
dead=3     ;BROWN
complx=2   ;RED
cancer=1   ;GREEN
dormant=4   ;YELLOW

nt = ntimestep+1ULL
sumntotal1  = LONARR(nt)
sumnc1      =  LONARR(nt)
ssumr1      = LONARR(nt)
ssumd1      = LONARR(nt)
ssumnd1     = LONARR(nt)
ssumdormant = LONARR(nt)
ssumcomplx1 = LONARR(nt)
num_inner   = LONARR(nt)
num_ibetween= LONARR(nt)
num_ioutter = LONARR(nt)

pop = LONARR(ilx+3L,ilx+3L)
pop_shw = LONARR(ilx+3L, ilx+3L)
```

```
dxyz = icom

x = ULONARR(dxyz)
y = ULONARR(dxyz)
z = ULONARR(dxyz)

;!(1)---------   Set position (i,j,k) of each site in array a  -------

ii =0L
for i = 0L,ilx-1 do begin
for j = 0L,iLx-1 do begin
for k = 0L,iLx-1 do begin

x[ii] = i
y[ii] = j
z[ii] = k
ii = ii +1L

endfor
endfor
endfor
;!(1)-----------------------------------------------------------------

;{(2)  LOOP DO RUN

for  ncolony = 1,num do begin

;!----------------------   Initial array a   ----------------------

 a = ULONARR(dxyz)
 b = ULONARR(dxyz)

;!--  set the first five cancer cell and input details with -----
;!--------  effecter of the middle of lattice  ----------------

imd   =  (icom-1ULL)/2ULL
;print, '  middle index',imd
a[imd]       = cancer
a[imd-1]     = cancer
a[imd+1]     = cancer
a[imd-iLx]   = cancer
```

```
a[imd+iLx]  = cancer
a[imd-iLx*iLx]  = cancer
a[imd+iLx*iLx]  = cancer


for i = 0L,icom-1L do begin
icol = z[i]
irow = y[i]
pop[icol,irow]=a[i]
endfor

window, 0, xsize = ixy_shw, ysize = ixy_shw
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)


ishw =1

case ishw of
1:begin
;Define :show pop === population in the middle cut x
for i = ((iLx-1L)/2L)*ilx*ilx, ((iLx+1L)/2L)*ilx*ilx do begin
      iyyy = z[i]
      ixxx = y[i]
      pop[1,1]=complx
      pop[2,1]=dead
      pop[3,1]=cancer
      pop[4,1]=dormant
      pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
  end

2:begin
  ;Define :show pop === population in the middle cut y
  for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
      iyyy = x[i]
      ixxx = z[i]
      pop[1,1]=complx
      pop[2,1]=dead
      pop[3,1]=cancer
```

```
        pop[4,1]=dormant
        pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
  end

3: begin
;Define :show pop === population in the middle cut z
for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
        iyyy = x[i]
        ixxx = y[i]
        pop[1,1]=complx
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
        pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
end
0:begin
end
endcase




;!----------  initial: array ichosen, ichosen3,iguess   -------

ichosen     =ULONARR(icom)
ichosen_up =ULONARR(icom)
iguess      =ULONARR(10ULL)
icount=0ULL

;!--------     FOR COUNTING AND REFERENCE AT THE FIRST TUMOR CELLS    --------

ichosen[1]=imd
ichosen[2]=imd-1ULL
ichosen[3]=imd+1ULL
ichosen[4]=imd-iLx
```

```
ichosen[5]=imd+iLx
ichosen[6]=imd-iLx*iLx
ichosen[7]=imd+iLx*iLx

;!------------------------        Set initial      ----------------

nc=7L
nd=0L
nee=0L
ndor=0L
ntot=7L
ntotal=7L
seed = 107L

sx=0.0d
sy=0.0d
sz=0.0d
sumr=0.0d
r=0.0d
d=0.0d

b = a

;(3){{  DO TIMESTEPS IN A COLONY
     for nday =1,ntimestep do begin
          ntot=ntotal
          ntotal=0ULL

 ;input logistic growth function
          rkup=akst*(1d - (nc*1d/iphi*1d))
          rkk = aktwo*(1d - (nee/iphi*1d))

;{{{(4)   LOOP A TIME STEP

          for iik=1ULL,ntot do begin
          ord=0

;CONDITION TO ORDERING

           index = ichosen[iik]
```

```
;CHECKING ------------------------------------
            if (a[index] eq 0) then print, 'error'
    ;       [                   if case E                    ]
            ndo=0
            if  ( (a[index] eq complx) and (ndo eq 0) ) then begin
                ndo=1
                ntotal = ntotal +1ULL
                ichosen_up[ntotal]=index
;!----------       If random number less than and equal k3----------
                r =randomu(seed)
                if ( r le akrd) then begin
                  nee=nee-1ULL
                  nd=nd+1ULL
                  b[index]=dead
                endif else begin
                if(  r gt (1-akndrev)) then begin
                  nee=nee-1ULL
                  nc = nc +1ULL
                  b[index]=cancer
                endif
                endelse
            endif
;       [                endif case E                    ]


;       [                  if case D                     ]
;
            if ( (a[index] eq dead)  and (ndo eq 0) ) then begin
                    ndo=1
                r =randomu(seed)
                if (r  le  akfour ) then begin
                  nd=nd-1ULL
                  b[index]=normal
                endif else begin
                  ntotal = ntotal +1ULL
                  ichosen_up[ntotal]=index
                endelse
                if (r gt (1 - akfive)) then begin
                  ndor=ndor+1ULL
                  b[index]=dormant
                  ntotal=ntotal + 1ULL
```

```
                    ichosen_up[ntotal]=index
                  endif
               endif

;       [                End if case D                    ]
;           [           if case dormant                     ]

              if ( (a[index] eq dormant)  and (ndo eq 0) ) then begin
                    ndo=1
                    ntotal = ntotal +1ULL
                    ichosen_up[ntotal]=index
                    b[index]=dormant

               endif


;       [                End if case Dormant                    ]

;         [                  if case Cancer                     ]
              if ( (a[index] eq cancer) and (ndo eq 0) ) then begin

                  ndo=1
                  ntotal = ntotal +1ULL
                  ichosen_up[ntotal]=index

;!-------------------------k_1prime-----------------------------
                  ranfix=randomu(seed)
                  icount = 0
;!=======================Case one of Cancer====================
;!-------------------------k1 prime---------------------------
                  if(ranfix le rkup) then begin
;!----------------      Average radius-(R)and Density (D) ----------

                     if ( (a[index-1] eq 0) and (b[index-1] eq 0) ) then begin
                      icount = icount +1ULL
                      iguess[icount]=index-1ULL
                     endif
                     if ( (a[index+1] eq 0) and (b[index+1] eq 0) ) then begin
                      icount = icount +1ULL
                      iguess[icount]=index+1
                     endif
```

```
                    if( (a[index-ilx] eq 0)  and (b[index-ilx] eq 0))then begin
                     icount = icount +1ULL
                     iguess[icount]=index-ilx
                    endif
                    if( (a[index+ilx] eq 0) and (b[index+ilx] eq 0)) then begin
                      icount = icount +1ULL
                     iguess[icount]=index+ilx
                    endif
                    if((a[index-ilx*ilx] eq 0) and (b[index-ilx*ilx] eq 0)) then begin
                     icount = icount +1ULL
                     iguess[icount]=index-ilx*ilx
                   endif
                   if((a[index+ilx*ilx] eq 0) and (b[index+ilx*ilx] eq 0))then begin
                      icount = icount +1ULL
                      iguess[icount]=index+ilx*ilx
                   endif
                     if(icount ge 1) then begin
                      nc=nc+1L
                      icho=1+fix(icount*randomu(seed))
                      b[iguess[icho]]=cancer
                      ntotal = ntotal +1ULL
                      ichosen_up[ntotal]=iguess[icho]
                     endif



                  ;endif


;!=======================Case two of cancer=======================
;!-------------------------cancer => complex---------------------

                  endif else begin
                     if ((ranfix gt rkup) and (ranfix ge (1-rkk) )) then begin
                       b[index]=complx
                       nc = nc-1ULL
                       nee = nee+1ULL
                     endif
                   endelse
            endif
;         [                End  if case C                    ]
```

```
      endfor
;===========UPDATE LATTICE
          ichosen = ichosen_up
          a = b
;(4)  End loop for one day  }}}
;!----------------Average radius-(R)and Density (D) ----------

;=========================Show everday or show last day


if (nday eq ntimestep) then begin
for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
        iyyy = z[i]
        ixxx = y[i]
        pop[1,1]=complx
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
         pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
endif
;                          end loop   Show everyday

;               START CALULCUTION ZONE

     sx=0.0
     sy=0.0
     sz=0.0
     sumr=0.0
     r=0.0

;------------ calculation radius & density
for i=0ULL,icom-1ULL do begin
 if( a[i] ne normal ) then begin
 sx=x[i]-x[imd]
 sy=y[i]-y[imd]
 sz=z[i]-z[imd]
 sumr = sumr + long(sqrt(sx*sx+sy*sy+sz*sz))
 endif
endfor
```

```
;            end loop calculation

r=sumr/double(nc+nee+nd+ndor)
d=(double(nc+nee+nd+ndor))/(r*r*r)
;---------- calculation radius & density

 ;--------- count cancer in each region
 inner = 0d
 ibetween =0d
 ioutter =0d
 dis = 0.0
 for i=0ULL,icom-1ULL do begin
            if (a[i] eq cancer) then begin
               sx=x[i]-x[imd]
               sy=y[i]-y[imd]
               sz=z[i]-z[imd]
               dis= sqrt(sx*sx+sy*sy+sz*sz)

                if (dis lt r/2) then begin
                      inner = inner +1ULL
                endif else begin
                   if (dis lt 0.8*r) then begin
                     ibetween = ibetween +1ULL
                   endif  else begin
                     ioutter = ioutter +1ULL
                   endelse
                endelse
             endif
endfor
;  end of loop  count cancer in each region


 print,r, nc,float(inner)/float(nc), float(ibetween)/float(nc), float(ioutter)/float(nc)
 ;endif


;
;==============keep value every day
          num_inner[nday] = num_inner[nday]+inner
          num_ibetween[nday] =  num_ibetween[nday]+ibetween
```

```
          num_ioutter[nday] =  num_ioutter[nday]+ioutter
          sumntotal1[nday] = sumntotal1[nday] + double(nc+nee+nd)
          sumnc1[nday]     = sumnc1[nday]     + double(nc)



          ssumr1[nday]     = ssumr1[nday]     + r
          ssumd1[nday]     = ssumd1[nday]     + d
          ssumnd1[nday]    = ssumnd1[nday]    + nd
          ssumcomplx1[nday] = ssumcomplx1[nday]  + nee
          ssumdormant[nday]=ssumdormant[nday]+ndor
     endfor

;(3)  End loop for one colony in ntimestep days  }}

;======== Set 1
  if (ncolony mod num) eq 0 then begin

       printf,18,'---------------------------------------r two rev equ',aktwo
  for i = 1,ntimestep do begin
    printf,18,i,sumntotal1[i]/float(num),sumnc1[i]/float(num),ssumcomplx1[i]/float(num),ssumdormant[i]/float(num)
   endfor
;======= measure and count
  printf,18,'---------------------------------------'
   for i = 1,ntimestep do begin
   printf,18,i,ssumr1[i]/float(num),num_inner[i]/float(num),num_ibetween[i]/float(num),$
    num_ioutter[i]/float(num)
   endfor

 t1 = systime()
 printf,18,t0,t1
 PRINT,t0,t1




;====================For Graph
t=indgen(nt)
t1=indgen(nt)
test2 = fltarr(nt)
test1 = fltarr(nt)
test3 = fltarr(nt)
```

```
test4= fltarr(nt)
testr1 = fltarr(nt)
testr2 = fltarr(nt)
testr3 = fltarr(nt)
testr4 = fltarr(nt)
;==============I-T
testimm1 =fltarr(nt)
testimm2=fltarr(nt)

test2 = sumntotal1/float(num)
test4 = sumnc1/float(num)
testr2 = ssumr1/float(num)
testr3 =ssumd1/float(num)

;============IMMUNE-TUMOR
testimm1 = ssumcomplx1/float(num)



for i = 0,ntimestep-1 do begin
    t1[i]= t[i+1]
    test1[i] = test2[i+1]
    test3[i] = test4[i+1]
    testr1[i] = testr2[i+1]
    testr4[i] = testr3[i+1]
    testimm2[i] = testimm1[i+1]
endfor
yxz = 1

;show graph after run ============================
if (yxz eq 1) then begin
;============Window 5 shows N, Nc, R
window, 5, xsize = nx, ysize = ny
plot, t1, test1,TITLE ='N & R & Nc'
oplot, t1, test3,LINESTYLE = 1
oplot,t1,testr1*38.00,LINESTYLE = 3


;==========Window 6 shows R and density
window, 6, xsize = nx, ysize = ny
plot, t1,testr1,TITLE=' R and density '
```

```
oplot, t1,testr4,LINESTYLE = 1


;==========Window 7 shows Nc and E====
window, 7, xsize = nx, ysize = ny
plot, t1,test3,TITLE=' Tumor-Immune  '
oplot, t1,testimm2,LINESTYLE = 4


endif
;show graph after run ============================


endif
;====Print on file


print,ncolony


endfor
; ======================  end loop for vary k1


;
endfor
;(2)  End do runs   }

close, 18
end
```

3.2.2 ตัวโปรแกรม ชื่อ cancer_immune_t1000

```
pro cancer_immune_t1000
```

```
;Run June 15 2010 step r1
;run r2 for 0,0.1,0.3
;SETTING RUN NUMBER AND TIME STEP AND INITIAL SEED NUMBER

num=1
ntimestep = 250

seed = 107L

t0 = systime()

  openw, 18, 'cancer_immune_t1000.txt'
;SETTING PARAMETERS
akst=0.5  ;r_prolif
aktwo=0.05   ;r_binding
akndrev=0.1 ;r_detach
akrd= 0.4    ;r_lysis
akfour=0.5  ;r_decay
akfive=0.3   ;r dormant
iphi= 100000ULL


;                LOOP  FOR RUN VARY EACH RATE              THE  BIGGEST LOOP

for iloo = 1, 1 do begin
akst  =  akst + 0.05*(iloo-1)
;               step*0.1
;=====================================================================

;SETTING THE LATTICE SIZE (MAX 301 X 301 X 301 SITES)

iLx=251ULL
iLy=iLx
icom= iLx*iLx*iLx
print, '   THE TOTAL SITES = ',icom
```

```
;SETTING SIZE OF LATTICE SHOWING

device, retain=2, decomposed=0
loadct,4

ixy_shw=500   ;SIZE OF SHOWING LATTICE
window, 0, xsize = ixy_shw, ysize = ixy_shw


;SHOW CUT X-PLANE Y-PLANE OR Z-PLANE
ishw = 1;(shw =1  cutx);ishw = 2;(shw cuty);ishw = 3;(shw cutz)
nx = 500
ny = 500

;SETTING CELLS TYPE VARIABLES

normal=0   ;BLACK
dead=3     ;YELLOW
complx=2   ;RED
cancer=1   ;GREEN
dormant=4   ;what 's color?


nt = ntimestep+1ULL
sumntotal1  = LONARR(nt)
sumnc1      =  LONARR(nt)
ssumr1      = LONARR(nt)
ssumd1      = LONARR(nt)
ssumnd1     = LONARR(nt)
ssumdormant = LONARR(nt)
ssumcomplx1 = LONARR(nt)
num_inner   = LONARR(nt)
num_ibetween= LONARR(nt)
```

```
num_ioutter = LONARR(nt)

pop = LONARR(ilx+3L,ilx+3L)
pop_shw = LONARR(ilx+3L, ilx+3L)

dxyz = icom

x = ULONARR(dxyz)
y = ULONARR(dxyz)
z = ULONARR(dxyz)

;!(1)---------   Set position (i,j,k) of each site in array a  -------

ii =0L
for i = 0L,ilx-1 do begin
for j = 0L,iLx-1 do begin
for k = 0L,iLx-1 do begin

x[ii] = i
y[ii] = j
z[ii] = k
ii = ii +1L

endfor
endfor
endfor
;!(1)-----------------------------------------------------------------

;{(2)  LOOP DO RUN

for  ncolony = 1,num do begin

;!----------------------   Initial array a   ----------------------

 a = ULONARR(dxyz)
```

```
 b = ULONARR(dxyz)

;!--  set the first five cancer cell and input details with -----
;!--------  effecter of the middle of lattice  ----------

imd   =  (icom-1ULL)/2ULL
;print, '  middle index',imd
a[imd]       = cancer
a[imd-1]     = cancer
a[imd+1]     = cancer
a[imd-iLx]   = cancer
a[imd+iLx]   = cancer
a[imd-iLx*iLx]   = cancer
a[imd+iLx*iLx]   = cancer


for i = 0L,icom-1L do begin
icol = z[i]
irow = y[i]
pop[icol,irow]=a[i]
endfor

window, 0, xsize = ixy_shw, ysize = ixy_shw
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)


ishw =1

case ishw of
1:begin
;Define :show pop === population in the middle cut x
for i = ((iLx-1L)/2L)*ilx*ilx, ((iLx+1L)/2L)*ilx*ilx do begin
        iyyy = z[i]
        ixxx = y[i]
        pop[1,1]=complx
```

```
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
        pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
  end

2:begin
  ;Define :show pop === population in the middle cut y
  for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
        iyyy = x[i]
        ixxx = z[i]
        pop[1,1]=complx
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
        pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
  end

3: begin
;Define :show pop === population in the middle cut z
for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
        iyyy = x[i]
        ixxx = y[i]
        pop[1,1]=complx
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
        pop[ixxx,iyyy]=a[i]
endfor
```

```
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
print,'show at ',ishw
end
0:begin
end
endcase



;!----------  initial: array ichosen, ichosen3,iguess   -------

ichosen     =ULONARR(icom)
ichosen_up =ULONARR(icom)
iguess      =ULONARR(10ULL)
icount=0ULL

;!--------    FOR COUNTING AND REFERENCE AT THE FIRST TUMOR CELLS   --------

ichosen[1]=imd
ichosen[2]=imd-1ULL
ichosen[3]=imd+1ULL
ichosen[4]=imd-iLx
ichosen[5]=imd+iLx
ichosen[6]=imd-iLx*iLx
ichosen[7]=imd+iLx*iLx

;!-----------------------        Set initial      ----------------

nc=7L
nd=0L
nee=0L
ndor=0L
ntot=7L
ntotal=7L
```

```
seed = 107L

sx=0.0d
sy=0.0d
sz=0.0d
sumr=0.0d
r=0.0d
d=0.0d

;---------- calculation radius & density
for i=0ULL,icom-1ULL do begin
 if( a[i] ne normal ) then begin
 sx=x[i]-x[imd]
 sy=y[i]-y[imd]
 sz=z[i]-z[imd]
 sumr = sumr + long(sqrt(sx*sx+sy*sy+sz*sz))
 endif
endfor
r=sumr/double(nc+nee+nd+ndor)
d=(double(nc+nee+nd+ndor))/(r*r*r)
;---------- calculation radius & density
b = a

;(3){{  DO TIMESTEPS IN A COLONY
     for nday =1,ntimestep do begin
          ntot=ntotal
          ntotal=0ULL

 ;input logistic growth function
          rkup=akst*(1d - (nc*1d/iphi*1d))
          rkk = aktwo*(1d - (nee/iphi*1d))

;{{{(4)   LOOP A TIME STEP

          for iik=1ULL,ntot do begin
```

```
          ord=0
if( ord eq 1) then begin
;SET NO REORDERING
 if (iik eq NTOT+1) then begin
;---------------------------------------------------------------
;LOOP FOR RE-ODRERING ; nba ===> mmr_1 is the re-ordered array
mmr=ULLARR(ntot)
mmr_1=ULLARR(ntot+1ULL)
;Loop for re-ordering the sequence of label number of cell
       nba = indgen(ntot)+1
       for nb=0,ntot-1 do begin
         ntemp=ntot-nb
         r=randomu(iseed)
         nr=fix(r*ntemp)
         mmr(nb)=nba(nr)
         if(nr lt ntemp) then begin
           for jb=nr+1,ntemp-1 do begin
             nba(jb-1)=nba(jb)
           endfor
         endif
       endfor
;Shift the label
for i=0,ntot-1 do begin
 bjj=i+1
 mmr_1[bjj]=mmr[i]
endfor
endif
;---------------------------------------------------------------
;END OF LOOP FOR RE-ORDERING
               inew=mmr_1[iik]
               index=ichosen[inew]
endif

;CONDITION TO ORDERING
```

```
          index = ichosen[iik]

;CHECKING ---------------------------------
          if (a[index] eq 0) then print, 'error'
    ;     [                  if case E                    ]
          ndo=0
          if  ( (a[index] eq complx) and (ndo eq 0) ) then begin
              ndo=1
              ntotal = ntotal +1ULL
              ichosen_up[ntotal]=index
;!----------      If random number less than and equal k3----------
              r =randomu(seed)
              if ( r le akrd) then begin
                nee=nee-1ULL
                nd=nd+1ULL
                b[index]=dead
              endif else begin
              if(  r gt (1-akndrev)) then begin
                nee=nee-1ULL
                nc = nc +1ULL
                b[index]=cancer
              endif
              endelse
          endif
;       [                endif case E                   ]


;       [                  if case D                    ]
;
          if ( (a[index] eq dead)  and (ndo eq 0) ) then begin
                ndo=1
              r =randomu(seed)
              if (r  le  akfour ) then begin
                nd=nd-1ULL
                b[index]=normal
```

```
                endif else begin
                  ntotal = ntotal +1ULL
                  ichosen_up[ntotal]=index
                endelse
                if (r gt (1 - akfive)) then begin
                  ndor=ndor+1ULL
                  b[index]=dormant
                  ntotal=ntotal + 1ULL
                  ichosen_up[ntotal]=index
                endif
              endif

;       [                End if case D                    ]
;           [           if case dormant                     ]

             if ( (a[index] eq dormant)  and (ndo eq 0) ) then begin
                    ndo=1
                    ntotal = ntotal +1ULL
                    ichosen_up[ntotal]=index
                    b[index]=dormant

              endif


;       [                End if case Dormant                    ]

;         [                  if case Cancer                    ]
              if ( (a[index] eq cancer) and (ndo eq 0) ) then begin

                    ndo=1
                    ntotal = ntotal +1ULL
                    ichosen_up[ntotal]=index

;!-------------------------k_1prime-----------------------------
                    ranfix=randomu(seed)
```

```
                    icount = 0
;!=======================Case one of Cancer====================
;!-------------------------k1 prime--------------------------
                    if(ranfix le rkup) then begin
;!----------------      Average radius-(R)and Density (D) ----------

                      if ( (a[index-1] eq 0) and (b[index-1] eq 0) ) then begin
                       icount = icount +1ULL
                       iguess[icount]=index-1ULL
                      endif
                      if ( (a[index+1] eq 0) and (b[index+1] eq 0) ) then begin
                       icount = icount +1ULL
                       iguess[icount]=index+1
                      endif
                      if( (a[index-ilx] eq 0)  and (b[index-ilx] eq 0))then begin
                       icount = icount +1ULL
                       iguess[icount]=index-ilx
                      endif
                      if( (a[index+ilx] eq 0) and (b[index+ilx] eq 0)) then begin
                        icount = icount +1ULL
                       iguess[icount]=index+ilx
                      endif
                      if((a[index-ilx*ilx] eq 0) and (b[index-ilx*ilx] eq 0)) then begin
                       icount = icount +1ULL
                       iguess[icount]=index-ilx*ilx
                    endif
                    if((a[index+ilx*ilx] eq 0) and (b[index+ilx*ilx] eq 0))then begin
                       icount = icount +1ULL
                       iguess[icount]=index+ilx*ilx
                    endif
                      if(icount ge 1) then begin
                       nc=nc+1L
                       icho=1+fix(icount*randomu(seed))
                       b[iguess[icho]]=cancer
                       ntotal = ntotal +1ULL
```

```
                    ichosen_up[ntotal]=iguess[icho]
                   endif



                ;endif


;!=======================Case two of cancer=======================
;!-------------------------cancer => complex--------------------

                endif else begin
                   if ((ranfix gt rkup) and (ranfix ge (1-rkk) )) then begin
                     b[index]=complx
                     nc = nc-1ULL
                     nee = nee+1ULL
                   endif
                endelse
             endif
;         [                End  if case C                    ]
      endfor
;===========UPDATE LATTICE
         ichosen = ichosen_up
         a = b
;(4)  End loop for one day  }}}
;!----------------Average radius-(R)and Density (D) ----------


;========================Show everday or show last day


if (nday eq ntimestep) then begin
for i = ((iLx-1ULL)/2ULL)*ilx*ilx, ((iLx+1ULL)/2ULL)*ilx*ilx do begin
       iyyy = z[i]
       ixxx = y[i]
       pop[1,1]=complx
```

```
        pop[2,1]=dead
        pop[3,1]=cancer
        pop[4,1]=dormant
         pop[ixxx,iyyy]=a[i]
endfor
tvscl, congrid(pop(*,*),ixy_shw,ixy_shw)
endif
;                          end loop   Show everyday

;               START CALULCUTION ZONE

     sx=0.0
     sy=0.0
     sz=0.0
     sumr=0.0
     r=0.0

;------------ calculation radius & density
for i=0ULL,icom-1ULL do begin
 if( a[i] ne normal ) then begin
 sx=x[i]-x[imd]
 sy=y[i]-y[imd]
 sz=z[i]-z[imd]
 sumr = sumr + long(sqrt(sx*sx+sy*sy+sz*sz))
 endif
endfor

;           end loop calculation

r=sumr/double(nc+nee+nd+ndor)
d=(double(nc+nee+nd+ndor))/(r*r*r)
;------------ calculation radius & density

 ;--------- count cancer in each region
 inner = 0d
```

```
 ibetween =0d
 ioutter =0d
 dis = 0.0
 for i=0ULL,icom-1ULL do begin
            if (a[i] eq cancer) then begin
               sx=x[i]-x[imd]
               sy=y[i]-y[imd]
               sz=z[i]-z[imd]
               dis= sqrt(sx*sx+sy*sy+sz*sz)

                if (dis lt r/2) then begin
                      inner = inner +1ULL
                  endif else begin
                     if (dis lt 0.8*r) then begin
                       ibetween = ibetween +1ULL
                     endif  else begin
                       ioutter = ioutter +1ULL
                     endelse
                  endelse
             endif
endfor
;  end of loop  count cancer in each region


 print,r, nc,float(inner)/float(nc), float(ibetween)/float(nc), float(ioutter)/float(nc)
 ;endif



;
;=============keep value every day
          num_inner[nday] = num_inner[nday]+inner
          num_ibetween[nday] =  num_ibetween[nday]+ibetween
          num_ioutter[nday] =  num_ioutter[nday]+ioutter
          sumntotal1[nday] = sumntotal1[nday] + double(nc+nee+nd)
          sumnc1[nday]     = sumnc1[nday]     + double(nc)
```

```
          ssumr1[nday]      = ssumr1[nday]      + r
          ssumd1[nday]      = ssumd1[nday]       + d
          ssumnd1[nday]     = ssumnd1[nday]      + nd
          ssumcomplx1[nday] = ssumcomplx1[nday]  + nee
          ssumdormant[nday]=ssumdormant[nday]+ndor
     endfor

;(3)  End loop for one colony in ntimestep days  }}

;======== Set 1
  if (ncolony mod num) eq 0 then begin

       printf,18,'---------------------------------------r two rev equ',aktwo
  for i = 1,ntimestep do begin

printf,18,i,sumntotal1[i]/float(num),sumnc1[i]/float(num),ssumcomplx1[i]/float(num),ssumdormant[i]/float(num)
   endfor
;======= measure and count
  printf,18,'---------------------------------------'
   for i = 1,ntimestep do begin
   printf,18,i,ssumr1[i]/float(num),num_inner[i]/float(num),num_ibetween[i]/float(num),$
    num_ioutter[i]/float(num)
   endfor

 t1 = systime()
 printf,18,t0,t1
 PRINT,t0,t1




;====================For Graph
t=indgen(nt)
```

```
t1=indgen(nt)
test2 = fltarr(nt)
test1 = fltarr(nt)
test3 = fltarr(nt)
test4= fltarr(nt)
testr1 = fltarr(nt)
testr2 = fltarr(nt)
testr3 = fltarr(nt)
testr4 = fltarr(nt)
;=============I-T
testimm1 =fltarr(nt)
testimm2=fltarr(nt)

test2 = sumntotal1/float(num)
test4 = sumnc1/float(num)
testr2 = ssumr1/float(num)
testr3 =ssumd1/float(num)

;===========IMMUNE-TUMOR
testimm1 = ssumcomplx1/float(num)



for i = 0,ntimestep-1 do begin
    t1[i]= t[i+1]
    test1[i] = test2[i+1]
    test3[i] = test4[i+1]
    testr1[i] = testr2[i+1]
    testr4[i] = testr3[i+1]
    testimm2[i] = testimm1[i+1]
endfor
yxz = 1

;show graph after run ============================
if (yxz eq 1) then begin
```

```
;============Window 5 shows N, Nc, R
window, 5, xsize = nx, ysize = ny
plot, t1, test1,TITLE ='N & R & Nc'
oplot, t1, test3,LINESTYLE = 1
oplot,t1,testr1*38.00,LINESTYLE = 3


;==========Window 6 shows R and density
window, 6, xsize = nx, ysize = ny
plot, t1,testr1,TITLE=' R and density '
oplot, t1,testr4,LINESTYLE = 1
;==========Window 7 shows Nc and E====
window, 7, xsize = nx, ysize = ny
plot, t1,test3,TITLE=' Tumor-Immune  '
oplot, t1,testimm2,LINESTYLE = 4


endif
;show graph after run ============================

endif
;====Print on file

print,ncolony


endfor
; ======================  end loop for vary k1

;
endfor
;(2)  End do runs   }

close, 18
end
```