Hybrid DNA and Enzymatic based Computation for Address Encoding, Link Switching and Error Correction in Molecular Communication

Frank Walsh[2], Sasitharan Balasubramaniam[1], Dmitri Botvich[1], Tatsuya Suda[3], Tadashi Nakano[3], Stephen F. Bush[4], Mícheál Ó Foghlú [1]

Telecommunication Software and Systems Group[1]
Waterford Institute of Technology[1, 2]
Carriganore Campus, Ireland
{sasib, dbotvich, mofoghlu}@tssg.org[1]
{fwwalsh}@wit.ie[2]
Information and Computer Science[3]
University of California, Irvine, CA, USA
{suda, tnakano}@ics.uci.edu
Computing and Decision Sciences[4]
GE Global Research
Niskayuna, NY, USA
bushsf@research.ge.com

ABSTRACT

This paper proposes a biological cell-based communication protocol to enable communication between biological based nano devices. Inspired by existing communication network protocols, our solution combines two molecular computing techniques (DNA and enzyme computing), to design a condensed three layer protocol stack for Molecular Communication Networks. Based on computational requirements of each layer of the stack, our solution specifies biomolecule address encoding, error correction and link switching mechanisms for molecular communication networks.

Keywords

Molecular communication, Molecular computation Communication protocols, Nano-computation.

1. INTRODUCTION

In common with networked computing devices, biological cells have the ability to transmit, receive and process information. Biochemical signaling networks and signal transduction mechanisms interact in a complex biochemical system that processes and reacts to chemically encoded information [7]. Just as modular components are used to compose electronic circuits, the mechanisms that underpin biochemical systems are now being investigated to create molecular components, where the aim is to engineer biological based nano (*bio-nano*) scale systems. One good example is the research areas of Molecular Computing [14], which manipulate biomolecules to engineer biochemical based computation systems. It is the fusion of Molecular Computing and Molecular Communication [1], a new research domain in supporting communication between bio-nano scale devices, which will provide the necessary computational mechanisms to create communication protocols for such devices. Just as data communication protocols resulted in the rapid growth and ubiquity of networked computing devices and applications, the development of communication protocols for nano-based networks will stimulate groundbreaking future applications of bio-nano devices. The potential applications of these combined technologies are vast, in particular in the medical field, where nano-scale devices can perform surgical procedures [16] or ensure accurate drug delivery to specific parts of organs and tissues.

Living cells contain various components that play a vital role in networked communication. These include, for example network interfaces (receptors, gap junctions), computation processes (regulatory networks, enzymatic signaling pathways) and memory capabilities (nucleic acids). In this paper, we propose a cell-based communication platform that uses these functional complexities to create protocols necessary for molecular communication networks. Our proposed hybrid solution, includes DNA as well as enzymatic based computation, where each contributes to a specific protocol function. We will describe how we will re-use protocols from communication networks, and transfer their mechanisms to a cell-based environment. In particular, we will show how we translate our condensed protocol stack to support addressing, error correction, and link switching. The paper is constructed as follows: Section 2 reviews related work. Section 3 investigates communication requirements for biochemical nano networks. Section 4 presents a simple connectionless communication solution using the living cell as a communication platform for address encoding, error correction, and link switching. Finally, section 5 presents conclusions and future work.

2. RELATED WORK

2.1 Molecular Communication

Molecular Communication uses encoded molecules as information carriers to engineer biochemical-based communication systems. In [11], Moritani et al define a Molecular Communication Interface that uses vesicles embedded with gap junction proteins to transport message-encoded molecules. Gap junctions form communication channels between cells and vesicles that allow small molecules such as ions, metabolites and small nucleotides to diffuse from cell cytoplasm to vesicle and visa versa. The vesicles act as signal carriers, propagating signal molecules between sender and receiver nano devices. The selectivity and permeability of a gap junction channel is affected

by various factors such as connexin phosphorylation[12], environmental pH and temperature. We will later describe how we use external control of selectivity and permeability properties to perform functions such as link switching, which one form of routing found in conventional data network devices [12].

2.2 Cell-based Computing

2.2.1 DNA Computing

In [3], Benenson et al present a programmable autonomous finite state automaton consisting entirely of biomolecules. The authors' design consists of a long DNA input molecule that is processed repeatedly by a restriction enzyme. Short DNA "rule" molecules control the operation of the restriction enzyme is precisely. This concept forms the basis for a nano scale computational machine that diagnoses disease and releases treatment molecules based on several disease-indicating inputs [13]. The authors predict that more complex machines such as stack automata and programmable DNA molecule-encoding applications will be developed using similar techniques. In [19], Liu et al extend the molecular automaton presented in [13] to design a "DNA-based Killer Automaton" that can release cytotoxic molecules which propagate to neighboring cells via gap junction channels.

DNA is the universal "information molecule" and an obvious choice for encoding information as a sequence of biochemical symbols. The now routine syntheses of custom DNA molecules combined with sophisticated software simulation tools are enabling increasingly complex DNA-based computation solutions.

2.2.2 Enzyme Computing

Another form of cell-based computation is enzyme computing. Markevich et al [4] have created a bistable switch using a cell-based Kinase-Phophatase signaling cascade (MAPK) that is highly conserved in eukaryotic cells. In doing so, the author demonstrates the use of ultra-sensitive cell-based enzyme signaling pathways as digital logic modules. Similarly, in [5] Stetter et al develop an enzyme reaction model for logical nano computation by manipulating the concentration of biological enzymes. Also using the bistable nature of biochemical enzymatic reactions, Stetter presents a reusable, "easy to engineer" architecture that forms the basis of several Boolean logic functions such as AND, and OR gates. The author demonstrates the practical application of the circuit using a cell-based Kinase-Phophatase reaction to implement a flip-flop circuit. This small enzyme-based circuit can act as a sub-component in composing more complex functions. However several challenges exist in creating complex *in-vivo* circuits such as chemical heterogeneity, uniformity and predictability [14]. Niazov et al [15] successfully orchestrated a series of interconnected logic gates based on similar enzyme reactions. To achieve modularization, each logic sub-unit must employ compartmentalizing mechanisms, for example a distinct chemical species set to prevent intrinsic chemical interference between gates or specificity providing scaffolding molecules [8]. These mechanisms can provide computational functions to support nano-scale computation for networked nano-devices.

There are a number of differences between the two types of cell-based computation, where each has certain disadvantages and advantages with respect to computation for communication protocols. Firstly, the computational complexity and speed associated with DNA computing is, as yet, not attainable using enzyme based computation [18]. Also, the parameter characterization effort required to achieve enzyme computation increases dramatically relative to circuit complexity [14]. This makes enzyme computing more suitable for relatively simpler circuits that require short computation time. On the other hand, DNA-based computation can support larger computing requirements. The other difference between enzyme and DNA based computing is that the reactions occur in the cell cytosol located near the cell membrane [7]. Therefore, this allows closer interaction with trans-membrane receptors and gap junctions. This makes it particularly suitable to simpler, responsive computation involving extra-cellular input and output.

3. DEFINING PROTOCOLS FOR MOLECULAR COMMUNICATION

In this section we will first describe the core characteristics of communication network protocols, and how these protocols will be re-used to support bio-nano devices.

3.1 Communication Network protocols

Communication networks provide protocols that exhibit the following properties; access mechanism to physical communication interface, encoding and addressing mechanism, error detection/correction techniques, and routing of packets between connected nodes. Physical interface controllers provide connection to physical transmission media and include mechanisms such as modulation and channel coding. The link layer functions manage access to the underlying physical layer, while flow control and acknowledgment mechanisms are usually implemented in higher layer protocols such as TCP. Communication can be connectionless or connection-oriented, where connectionless communication have lower data overhead, and are suitable for energy efficient networks such as wireless sensor networks. Another common protocol used in communication network is error correction, where techniques such as Forward Error Correction (FEC) can ensure that end devices can recover from any data corruption incurred during transmission. This could be through inclusion of redundancy in channel encoding process.

3.2 Protocols for Molecular Communication

As described earlier, our intention is to be able to re-use protocols from conventional communication networks for molecular communications. However, there are a number of factors that must be taken into account when translating communication network protocols to the environment of molecular communications. Firstly, propagation of information in molecular communication is typically characterized as low speed and in an environment where the link condition is highly variable compared to standard communication network [1][2]. These characteristics have repercussions for the design of protocols of molecular communication systems. Slow diffusion-based processes do not support the creation of high-speed switching functions common in conventional network devices that will require complex queuing mechanisms for packets. At the same time, due to high variability and harsh biological environment, the use of acknowledgements and retransmission of messages in the event of loss or corrupt packets may not lead to improved performance.

Due to power and size limitations, bio-nano devices could be designed with distinct sensing, processing and actuation characteristics that coordinate and cooperate via a molecular

communication network. We anticipate two types of information transmissions used in molecular communications, which includes sensory data (data collected from bio-nano devices) and command data (instructions for bio-nano devices). Therefore, the transmission mechanism and protocols to be used will be highly dependent on the nature of the information. For example, for sensor data, we may use single paths with UDP like transmissions with no error correction. However command information or high priority sensor data will be transmitted through redundant paths with error correction capabilities (e.g. FEC). Consideration will also have to be put towards the topology of networks for molecular communication, taking into consideration the need for routing, and scale of sensors. We propose the use of multiple overlapping ring topologies as shown in Fig. 1.

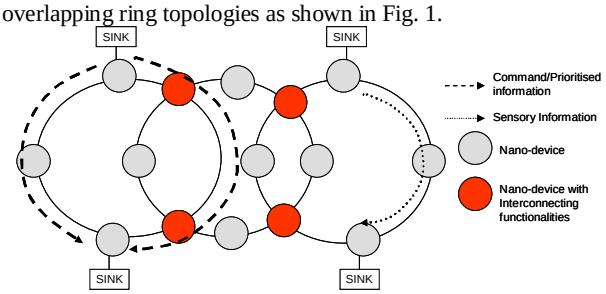


Figure 1. Ring based Network Topology for bio-nano devices

The reason for selecting the ring topology is to minimize the routing complexities that will be found in mesh networks. As shown in the diagram, the sensor information will only be required to be transmitted in single paths, while the commands or prioritized sensor information can be transmitted through redundant paths and also include error corrections if necessary. Since the topology will be static with pre-defined rate of traffic between the nodes, the routing tables can be static. In order to minimize the routing complexities, the routing process will only be performed at the interconnection nodes between different rings. Similarly, external interfaces with conventional communication devices for data collection and control purposes (i.e., information sink) will be provided at selected nodes to reduce functional complexity. Furthermore, as is the case in natural biochemical signaling processes, chemical messages are invariably "self-contained" molecules, not requiring segmentation or re-assembly of several received "packets". These physical characteristics indicate that a connectionless communication protocol is suitable for molecular communication, given the nature of the type of information to be transmitted. Therefore, it is vital that transmitted messages are decoded correctly at the receiver in the absence of connection-orientated properties.

Fig. 2 illustrates our condensed three-stack communication protocol for molecular communication. The stack includes Application, Transport, Network, and Physical layer. In our proposed protocol stack, we have deliberately left out the application layer and physical layer description. The application layer can be incorporated through interfacing with various types of nanomachines, while the physical layer transmission will be based on solutions by [1][12] for molecular communication. Since protocols can usually be defined through a Finite State Machine, we adopt a nano-logic circuit based computation to represent the different types of protocols. The nano-logic circuit is translated from the Finite State Machine representation of each layer of the protocol stack. As described in the related work, there are two main techniques of performing logic computation. Since, each technique has its own characteristics, we apply and select the right techniques based on two factors which includes, (i) location of computation within the cells (e.g. cytosol or nucleus), and (ii) complexity of computation. The enzymatic computation, due to their limited time requirement, is most suitable in performing small size logic circuit with high-speed computation.

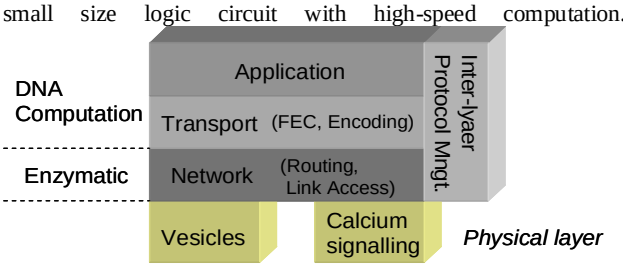


Figure 2. Protocol Stack for Molecular Communication

Therefore, this is most ideal for switching and routing (even though routing is done minimally, and only between rings) of information biomolecules between the nodes. The Transport and Application layer will require higher complexity computation and is usually not required to be time sensitive. Such computations will include FEC, addressing, and encoding/decoding. In between the two layers will be the Inter-layer protocol management, which will trigger the process of computation of the protocols and the location where this will happen in the cell.

In the next section, we introduce a nano-scale communication scenario based on these requirements. The proposed system specifies a physical molecular communication interface, link switching controller and address encoding mechanism that incorporates error correction.

4. PROPOSED SOLUTION

Fig. 3 illustrates our solution that combines the different processes for each protocol to support transmission for a single link.

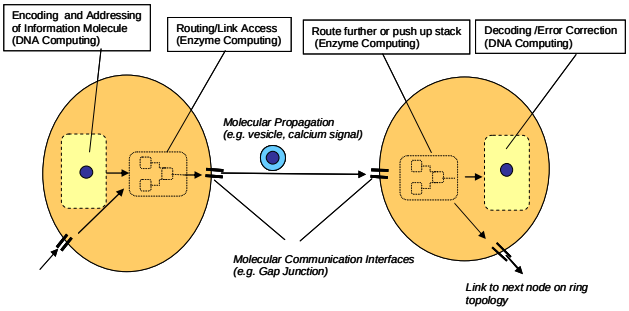


Figure 3. Mechanism of transmission for single link molecular communication

The communication process is as follows; initially, a nano-device encodes data as DNA biomolecules in the encoding compartment using a molecular encoding automaton. The encoded biomolecule is then further encoded with specific address of intended destination using an address table. Once this is complete, the transmission compartment switches the encoded molecule to the correct molecular communication interface link. At the physical layer, we adopt a suitable molecular communication mechanism such as [11] that uses vesicles to transport message molecules via gap junction hemi-channels. Other potential communication mechanisms include [12] that allow intercellular

communication using modulated calcium “waves” or [21] that specifies a molecular propagation system using natural biological motors such as kinesins.

We employ two molecular computing solutions to implement the protocols described, which are Benenson's DNA automata [3] and Stetter’s enzymatic computation for nano logic computation [5]. The DNA automata are used for message encoding, decoding and error correction, while the enzymatic computation techniques are used to switch between different link interfaces.

4.1.1 Encoding and Addressing

Fig. 4(a) illustrates the encoding solution. Similar to the model proposed by Liu et al in [19], our solution uses Benenson’s and Shiparo’s work in [3] to create a DNA-based automaton that produces molecules for intercellular communication. Each message is encoded as a unique sequence of nucleotide bases as demonstrated in [4]. For simplicity, only three addressable nano-device nodes are considered and each encoded message is ‘framed’ to include addressing information.

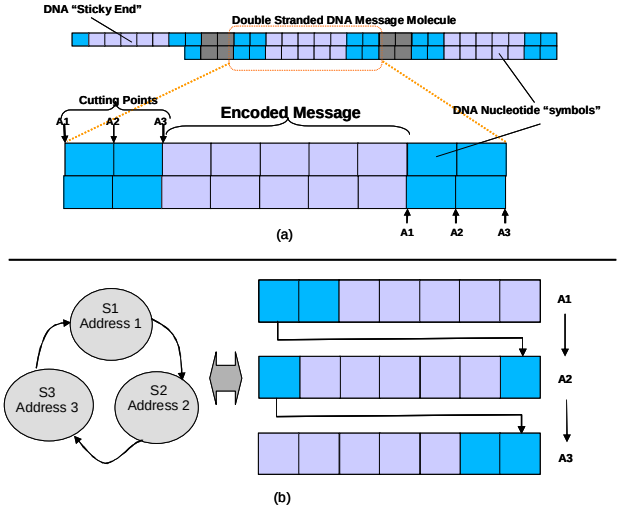


Figure 4. (a) Double Stranded DNA message molecule indicating restriction cut points for address encoding. (b) State representation of address encoding transitions.

Fig. 4(a) illustrates how nucleotide encoded messages are assembled in sequence of long input DNA message molecule with each message separated by a ‘spacer’ sequence. The upper leftmost “sticky end” represents the current state of the machine. During the address encoding process, the DNA message molecule is repeatedly cut by a restriction enzyme, which cuts off the leftmost segment of the molecule. Thus the *<address, message>* pairing represented by the current state of the encoding automaton is released as a bi-polymer segment through the restriction process. Fig. 4(b) illustrates how each address state and transition corresponds to actual encoded message molecule. Each state transition is enacted by a corresponding DNA “*rule*” molecule and enzyme complex that cleaves the corresponding nucleotide sequences. A key characteristic of computation is the precise cleavage of input message molecule that encodes or “frames” the message.

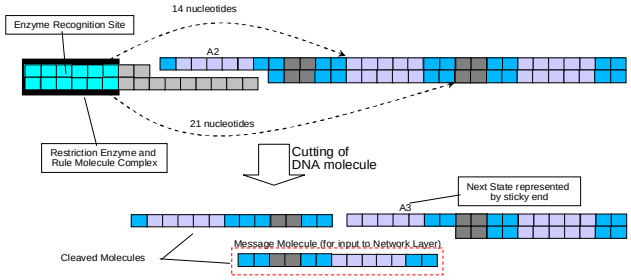


Figure 5. Mechanism of State Transition from Address 2 to Address 3 using Benenson’s Molecular Automata [3].

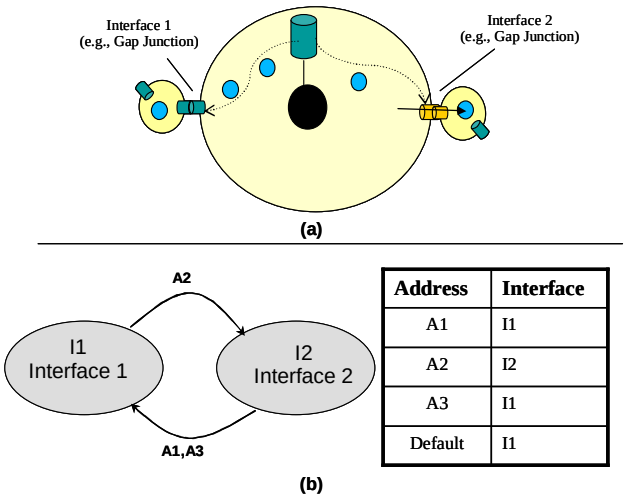
Fig.5 illustrates a rule execution transition from Address 2 to Address 3. Each rule molecule has a recognition site to which a restriction enzyme can bind. The number of nucleotide bases between the restriction enzyme and the sticky end of the rule molecule determines the precise locations of the message molecule cleave. In this example, the restriction enzyme complex combines with the message molecule and cuts at fourteen nucleotides on the top and twenty-one nucleotides at the bottom. The resulting new sticky end reveals the next state of the automaton. More importantly, the segment that is cut away is separated into two single strand DNA(ssDNA) molecules. The lower ssDNA molecule indicated in fig. 5 is the encoded message molecule with its rightmost end complementary to the new sticky end of the DNA message molecule.

Similar to techniques used in [19] and [13], the nanodevice can control computation by releasing molecules (e.g., mRNA) that selectively activate DNA “*rule*” molecules. [3] proposes that the molecules cleaved during computation can provide input to other parallel computational functions. In our solution, the cleaved ssDNA message molecules are released into the cytosol and provide the input to the molecular interface control function of the network layer. Theoretically, this mechanism can be extended to encode a multitude of unique address locations and any number of messages during computation.

4.1.2 Molecular Interface Control

Our study concerns the operation of a bio-nano device communication in a ring network. The ability to switch a message molecule to the correct communication interface based on message addressing is a key requirement. Fig. 6 illustrates a cell with two distinct molecular communication interfaces (e.g., distinct gap junctions). Each addressable location is switched through the corresponding interface according to the addressing state diagram shown in Fig. 6(b). For communication involving the transfer of message molecules through gap junctions, our solution is based on results provided in [20] which demonstrate the diffusion of synthetic oligonucleotides through gap junction channels.

In this study, interface selection is achieved using the “real world” implementation of the logical recurrent architecture as described by Stetter in [5]. The address-encoded molecule is input to the switching circuit which then releases/alters a corresponding chemical signal that “switches” the message to the correct interface. In the case of gap junction interfaces, the output of the enzyme-based circuit will therefore control the permeability of gap junction channels. Gap junction permeability is affected by connexin phosphorylation [12] via specific phosphorylation reagents, the concentration of which is controlled by the switching circuit.



Address	Interface
A1	I1
A2	I2
A3	I1
Default	I1

Figure 6. (a) Schematic diagram of cell with two distinct molecular communication interfaces (b). Address/Interface state diagram and switching table.

Thus Stetter's circuit can be used to effectively switch on and off each molecular communication interface by controlling the degree of phosphorylation of gap junction connexins. This in turn will allow the encoded message to be pushed through only a single link (or multiple links if multicasting is used). Using this technique, several communication links can be controlled simultaneously via compartmentalized enzymatic functions.

4.1.3 DNA Decoding and Forward Error Correction

Our decoding mechanism is also based on Benenson's DNA automata design. In [13] Benenson uses "protector strands" to control the operation of an enzyme based state machine by separating the constituent DNA strands of message molecules (see Fig. 7). In our solution, the protector strands are designed to have a strong affinity for received message molecules. Delivered message molecules cause the corresponding protector strand to separate from the transition strand and hybridize with the message molecule, allowing the formation, and thus activation, of a double stranded transition molecule, similar to the encoding process. The resulting transition molecule and restriction enzyme complex cleaves the corresponding decoding DNA molecule and releases the decoded DNA molecule.

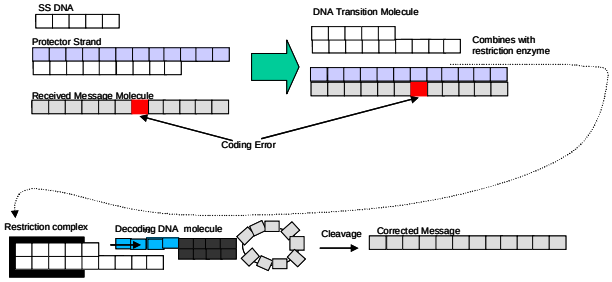


Figure 7. Forward Error Correction Mechanism

As already stated, prioritized messages require error detection and correction. Invariably, errors will occur in the encoding and transmission process of DNA molecules due to the imprecise nature of the associated complex biochemical reactions [17]. By including redundancy in the encoding process, error correction mechanisms can be incorporated into the decoding process. Our solution combines the nucleotide redundancy concept presented in [18] with DNA automata design in [13] to create an autonomous error correction mechanism. Each message molecule is composed of several repeated, identical nucleotide sequences. The hybridization of received message molecules by recognizing protector strands results in the successful release of the decoded message through cleavage of the decoder DNA molecule and the release of the output loop. Hybridization can still occur even though both single strand DNAs involved are not exactly complementary. Therefore once the message molecule is sufficiently "correct", it's delivery will instigate the separation of the protector strand from the transition molecule. This enables messages to be correctly decoded even though encoding errors exist in the message molecule. Thus simple Forward Error Correction is achieved using DNA automata. Fig. 7 also illustrates the cleaving of the decoding molecule that releases the output loop at the rightmost end of the molecule. The output loop is the corrected message molecule in this case.

5. CONCLUSION

Inspired by data communication protocols, we have presented a molecular communication protocol stack that successfully combines molecular computing and molecular communication techniques to provide a flexible cell-based bio-nano communication platform. We describe how the core characteristics of communication network protocols are re-used to design bio-nano device communication protocols. Our proposed solution presents the address encoding, link access, and error correction functions of a condensed protocol stack that are developed using suitable molecular computation techniques. Our solution demonstrates the necessity of matching the characteristics of each molecular computing technique to the computational requirements of each protocol stack layer. Our future work will investigate the feasibility of our design initially through simulation of chemical circuits for molecule encoding, decoding, link switching and error correction.

6. REFERENCES

[1] S. Hiyama, Y. Moritani, T. Suda, R. Egashera, A. Enomoto, M. Moore, T. Nakano, "Molecular Communication", Proc. NSTI Nanotech Conference and Trade Show. 3, pp.391-394, Anaheim, May 2005

[2] G. Alfano, D. Morandi, "On Information Transmission Among Nanomachines", 1st International Conference on Nano-Networks (Nano-Net), Lousanne, Switzerland, 2006.

[3] Y. Benenson, E. Shapiro, "Molecular Computing Machines", Dekker Encyclopaedia of Nanoscience and Nanotechnology, Marcel Dekker, 2004, pp. 2043 – 2055.

[4] N. Markevich, J. Hoek, B. Kholodenko "Signalling switches and bistability arising from multisite phosphorylation in protein kinase cascades" J. Cell Biology 164 (3): 353. (2004) http://www.jcb.org/cgi/reprint/164/3/353.

[5] M. Stetter, B. Schurmann, M. Hofstetter, "Logical Nano-Computation in Enzyme Reaction Networks" in Proceedings of BIONETICS 2006, Cavalese, Italy 2006.

[6] H. Sauro, B. Kholodenko "Quantitative analysis of signalling networks", Prog, in Biophysics & Mol. Biology, 86, 5-43, 2004.

[7] G. Cooper “The cell, a molecular approach”, ISBN 0-87893-119-8, pp545-549.

[8] N. Komarova, X. Zou, Q. Nie, L. Bardwell “A theoretical framework for specificity in cell signalling”, Mol Syst Biol. Oct. 2005.

[9] J.Reif, T.H.LaBean,”Autonamous Programmable Biomolecular Devices Using Self-Assembled DNA Nanostructures”, Communications of the ACM, vol.50, 9 pp.46-53, 2007.

[10] R. Weiss, “Cellular Computation and Communication using Engineered Genetic Regulatory Frameworks”, PhD Thesis, Massachusetts Institute of Technology, September 2001.

[11] Y. Moritani, S. Hyama, T. Suda “A Communication Interface Using Vesicles Embedded with Channel Forming Proteins in Molecular Communication”, in Proceedings of BIONETICS 2007, Budapest, Hungary 2007.

[12] T. Nakano, T. Suda, “Molecular Communication through Gap Junction Channels: System Design, Experiments and Modelling”, ” in Proceedings of BIONETICS 2007, Budapest, Hungary 2007.

[13] Y.Benenson, B. Gil, U. Ben-Dor, R. Adar, E. Shapiro. “An Autonomous Molecular Computer for Logical Control of Gene Expression”. Nature 429, pages 423–429, 2004.

[14] A. Presanna De Silva, S. Uchiyama, “Molecular Logic and Computing”, Nature Nanotechnology, vol. 2, July 2007.

[15] T. Niazof, R. Baron, E. Katz, O. Ioubashevski, I. Willner, “Concatenated logic gates using four coupled biocatalysts operating in series”, PNAS, Nov 14 2006, vol. {Niazov, 2006 #2}

[16] A. Cavalcanti, R. A. Freitas Jr., “Nanorobotics Control Design: A Collective behaviour approach for medicine”, IEEE Transaction on NanoBioscience, vol. 4, no. 2, June 2005

[17] H. Abelson, D.Allen, D.Coore, C.Hanson, G.Homsy, T. F. Knight,R.Nagpal, E.Rauch, G.Sussman, R.Weiss “Amorphous Computing”, Communications of the ACM, Volume 43, Number 5, May 2001

[18] L.Fedichkin, E.Katz,V.Privman, “Error Correction and Digitalization Concepts in Biochemical Computing”, J. Comput. Theor. Nanosci. 5, 36–43 (2008).

[19] S.Liu, J. Gaudiot, “DNA-based Killer Automaton: the Innovative Nanomedicine “, Proc. NSTI Nanotech Conference and Trade Show, Boston, Massachusetts, May 2006

[20] V. Valiunas, Y. Y. Polosina, H. Miller, I. A. Potapova, L. Valiuniene, S. Doronin, R. T. Mathias,R. B. Robinson, M. R. Rosen, I. S. Cohen, P. R. Brink, “Connexin-specific cell-to-cell transfer of short interfering RNA by gap junctions”, J Physiol 568.2 (2005) pp 459–468.

[21] S.Hiyama,Y.Moritani,T.Shima,T.Suda,K.Sutoh,"An Autonomous Molecular Transport System Using DNAs and Motor Proteins in Molecular Communication",in Proceedings of BIONETICS 2007, Budapest, Hungary 2007.