Int. J. Appl. Math. Comput. Sci., 2003, Vol. 13, No. 4, 585–591

SELF-REPLICATION PROCESSES IN NANOSYSTEMS OF INFORMATICS

STEFAN WĘGRZYN*, RYSZARD WINIARCZYK*, LECH ZNAMIROWSKI**

* Institute of Theoretical and Applied Informatics
Polish Academy of Sciences
ul. Bałtycka 5, 44–100 Gliwice, Poland
e-mail: ryswin@iitis.gliwice.pl

** Institute of Informatics, Silesian University of Technology
ul. Akademicka 16, 44–100 Gliwice, Poland
e-mail: lznamiro@top.iinf.polsl.gliwice.pl

Recent research on the nanotechnological processes of molecular products and object synthesis as well as research on the nanosystems of informatics, stimulates the development of technical systems of informatics. Until now, they have been used mainly for computational tasks when, similarly to biological organisms, they allowed the development of self-replicating products and complete objects. One can focus here on the model of a circulation of materials, information and energy in a biological cell, and a model of the self-replication phenomenon. In the model of materials, information and energy circulation, we can distinguish a multi-phase process of creating products which are later used for object construction. The first phase is a translation process based on information from mRNA, while the next one consists of post-translational modifications based on environmental interactions. The self-replication model presented in this paper refers to a self-replication of a material object together with its internal software, which had determined the object creation. The presented models of materials, information and energy circulation in a biological cell together with models of the self-replication phenomenon can be a basis for the design and development of technical systems of informatics capable of creating self-replicating products and objects.

Keywords: nanosystem of informatics, biological nanosystem of informatics, technical nanosystem of informatics, self-replication

1. Introduction

Generally speaking, a technology is a set of primitive elements and operations to be applied to those elements in order to obtain a final product or a result. If the primitive elements of a technology are single molecules and atoms, such a technology is called *nanotechnology*.

A system of informatics is defined by two parameters:

- the coding used for programs to be executed by the system,
- the hardware structure of the system which allows the programs to be executed automatically.

If the coding used in a given system of informatics uses single atoms and molecules as its terminal symbols, and if the operation of its hardware part relies on nanotechnology, such a system is called a *nanosystem of informatics* (Węgrzyn, 2001a).

One example of such nanosystems are the systems of informatics that are a part of all living organisms, and we shall refer to them as *biological nanosystems of informatics*.

In contrast to the biological systems of informatics, we shall determine the *technical nanosystems of informatics* as systems in which nanocoding and nanotechnological operations are implemented by technical means, without a direct use of cells of biological organisms.

As regards technological nanosystems of informatics, recent technological developments in nanocoding and nanotechnologies have formed the background for the creation of products and objects which self-replicate in a manner similar to that of biological systems.

2. The Metabolism of Materials, Energy and Information in a Cell of a Biological Organism

The life processes of a biological cell involve a metabolism of materials, energy and information controlled by programs coded in DNA, as shown in Fig. 1.

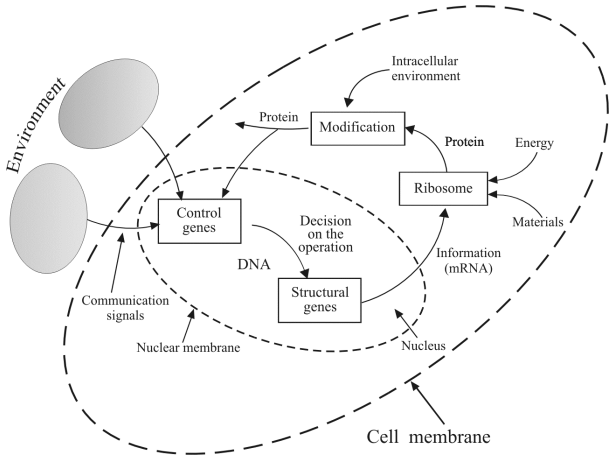


Fig. 1. Metabolism of materials, information and energy in a biological cell.

The manufacturing of materials needed by the cell is a result of a nanotechnological synthesis of molecules in successive processes of transcription, translation and post-translational modification (Znamirowski and Zukowska, 2002), as shown in Fig. 2. An example of post-translational modifications of proteins is presented in Fig. 3.

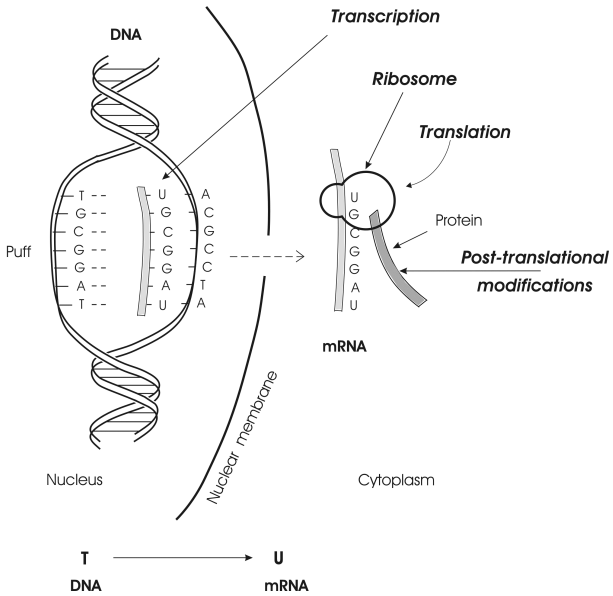


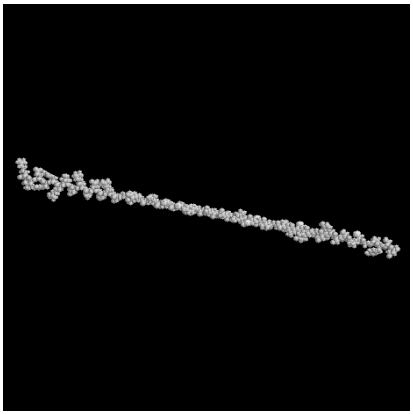
Fig. 2. Transcription, translation and post-translational modifications.

The processes occurring in the cell are controlled by programs coded in DNA and RNA, where the code uses nucleotides as its terminal symbols. In DNA the nucleotides involved are adenine (A), thymine (T), cytosine (C), and guanine (G). In RNA, thymine molecules (T) are replaced by uracil (U).

The necessary density of code requires a nanosystemic solution. For example, the length of the program stored in a single grain of wheat is estimated at many tens of megabits.

AAU	AGC	UAU	CCU	GGU	UGU
CCU	AGU	AGU	UAU	GAU	GGU
UAU	UGU	UUA	AAU	GGU	GGU
GUA	UGU	AUG	CAU	AUA	GAA
AGU	UUA	GAU	AGU	UAU	ACA
UGU	AAU	UGU	GUA	AUA	GGU
UAU	AGU	GGU	GAU	AGA	UGU
CAA	ACA	AGA	GAU	UUA	AGA
UGG	UGG	GAA	UUA	AGA	

(a)



(b)

(c)

Fig. 3. Post-translational modifications of the protein conformation: (a) the mRNA nucleotide chain coding the epidermal growth factor (EGF), (b) the RasMol representation of a straight protein EGF, (c) the spatial shape of a folded protein EGF (torsion angles (Liebecq, 1992) for glycine equal to 30^{o}, different zoom).

3. Post-Translational Modifications

3.1. Polypeptide Structure

The three-dimensional structure of a polypeptide can be completely described by placing it in a Cartesian coordinate system and listing the (x, y, z) coordinates for each atom in the chain. However, when we simulate modification processes, the synthesis programs generating the conformation of the main chain of the polypeptide use the amino acids library and need information on the angles between the α-carbons bonded with their side chains and contiguous peptide groups (Znamirowski, 2002). The backbone of a chain of amino-acid residues determined by the set of torsion angles is presented in Fig. 4.

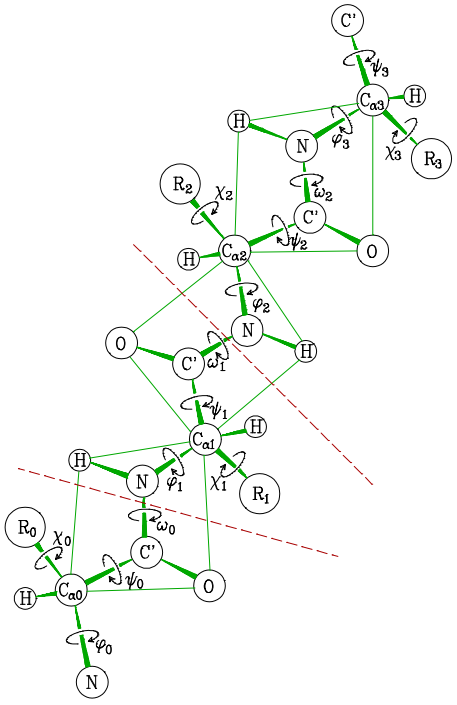


Fig. 4. Definition of protein torsion angles φ, ψ, ω and χ in the polypeptide sequence. The limits of the first residue are indicated by dashed lines, and the chain is shown in a fully extended conformation ($\varphi_1 = \psi_1 = \omega_1 = 180^\circ$) (Liebecq, 1992).

The backbone conformation of amino-acid residues can be specified by listing the torsion angles φ (rotation around the nitrogen-α-carbon bond in the main chain), and ψ (rotation around the α-carbon-carbon bond in the main chain of the polypeptide). The relationship between the peptide groups, α-carbons and torsion angles can be expressed in the following form:

$$\begin{aligned} &\rightarrow \mathrm{PB} \rightarrow \varphi_1 \rightarrow \mathrm{C}_{\alpha 1} \rightarrow \psi_1 \rightarrow \mathrm{PB} \\ &\qquad \rightarrow \varphi_2 \rightarrow \mathrm{C}_{\alpha 2} \rightarrow \psi_2 \rightarrow \mathrm{PB} \rightarrow, \end{aligned} \tag{1}$$

where PB denotes the peptide bond and $\mathrm{C}_{\alpha i}$ is the i-th α-carbon atom.

The zero for the torsion angle φ is defined with the N—H bond trans (Liebecq, 1992) to the C_α—C′ bond, and the zero position for ψ is defined with the C_α—N trans to the C═O bond. The peptide-bond torsion angle ω is generally 180°, this being with C═O bond trans to the N—H bond (Liebecq, 1992). A complete description of the spatial structure of a protein also requires the knowledge of the side-chain torsion angles χ.

The torsion angles play a crucial role in the conformation of proteins because the three-dimensional structure of a protein determines its biological functions. On the other hand, not all combinations of torsion angles are possible, as many of them lead to collisions between atoms in adjacent residues. The possible combinations of the φ and ψ angles that do not lead to a collision can be plotted on a Ramachandran map (Ramachandran and Sasisekharan, 1968; Hovmoller *et al.*, 2002). It can be simply observed that the small changes in the torsion angles cause fundamental changes in the conformation of the polypeptide in the case when the chain of amino acids is very long.

3.2. Post-Translational Conformations

When the translation process reaches the last, termination stage, the chain of amino-acid residues forming the polypeptide is inserted into an environment which is usually the fluid. This fluid (cytosol) can be treated in the first approximation as a solution. The polypeptide forms a chain of the molecules of residues connected together through the rotational bond realized by peptide bonds.

In this case (cf. Fig. 5) a possible conformation depends on the mutual interaction between the residues of amino acids and the molecules of the solution, e.g. hydrophobic or hydrophilic interactions. A tendency of the system chain of polypeptide and solution molecules in some neighborhood to get a minimal free energy (Güntert and Wüthrich, 2001) is a reason of forcing the changes. This can be called the free conformation of the polypeptide chain.

In the case when an external force influences the polypeptide chain (e.g. chaperons or enzymes), that extended system (chain, solution and enzyme) also tends to reach a minimal internal energy. This can be called the forced conformation. It is important that the internal bonds, e.g. disufide or hydrogen bonds, can interrupt the conformation transient state.

In Fig. 5 the two main groups of post-translational modifications are presented: firstly, the chemical modifications based on the exchange of side-chains in a polypeptide chain and cutting the selected amino acids, and sec-

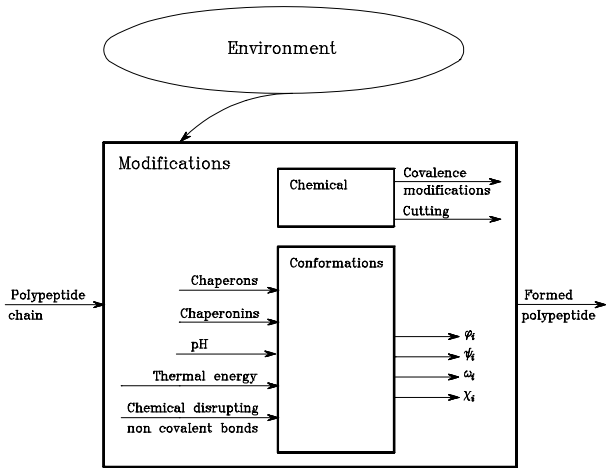


Fig. 5. Determination of the torsion angles as a result of the interaction of a polypeptide chain and the environment.

ondly, conformation modifications leading to determining the set of protein torsion angles φ_i, ψ_i, ω_i and χ_i in the amino-acid sequence.

3.3. Determining the Free Conformation by Dynamic Programming

Based on Fig. 4, we assume that the first amino-acid residue in the peptide chain appearing from the ribosome is labelled by 0, the next by 1 and so forth. Let us denote the potential energy function (Levitt *et al.*, 1995) of the zeroth amino-acid residue by E_0.

In the first approximation, we assume that the potential energy is a function of the torsion angles φ_0, ψ_0, ω_0 and χ_0. The shape of the backbone of the polypeptide chain depends only on the pairs of the angles φ and ψ, because ω usually equals $180^{\rm o}$ and χ is void.

As a consequence, the energy of the first residue of the peptide chain can be expressed as

$$E_0 = E_0(\varphi_0, \psi_0). \tag{2}$$

We quantize E_0 for $\varphi_0 = \{\varphi_0^0, \varphi_0^1, \ldots, \varphi_0^K\}$ and $\psi_0 = \{\psi_0^0, \psi_0^1, \ldots, \psi_0^L\}$. Therefore we have

$$E_0^{k,l} = E_0^{k,l}\left(\varphi_0^k, \psi_0^l\right).$$

The grid is not fully filled because of the Ramachandran restrictions.

When the residue of the next (i.e. the first) amino acid appears, the minimal energy of the bonded residues of amino acids 0 and 1 has the form

$$E_1^{k,l}\left(\varphi_1^k, \psi_1^l\right) := \min_{k,l}\left(E_0^{k,l} \Rightarrow E_1^{k,l}\right), \tag{3}$$

where the symbol '$\Rightarrow$' denotes the computation of the total energy at the point $E_1^{k,l}$ reached from points $E_0^{k,l}$ ($k = 1, 2, \ldots, K$ and $l = 1, 2, \ldots, L$), and the symbol '$:=$' denotes the computation of the right-hand-side expression (3) assigning the result to the left-hand side.

When the residue indexed as the second appears, the minimal energy for the three-bond residues of amino acids (at all accessible points on the grid) has the form

$$E_2^{k,l}\left(\varphi_2^k, \psi_2^l\right) := \min_{k,l}\left[\min_{k,l}\left(E_0^{k,l} \Rightarrow E_1^{k,l}\right) \Rightarrow E_2^{k,l}\left(\varphi_2^k, \psi_2^l\right)\right]. \tag{4}$$

When we reach the last n-th residue, we have

$$E_n^{k,l}\left(\varphi_n^k, \psi_n^l\right) := \min_{k,l}\left[\min_{k,l}\left(E_{n-2}^{k,l} \Rightarrow E_{n-1}^{k,l}\right) \Rightarrow E_n^{k,l}\left(\varphi_n^k, \psi_n^l\right)\right]. \tag{5}$$

This is the key point of the procedure. For $\min_{k,l} E_n^{k,l}(\varphi_n^k, \psi_n^l)$ in (5), when we get back to E_0, we can find an optimal "trajectory" (minimal energy) for the set of pairs (φ_i, ψ_i) in the form

$$(\varphi_n, \psi_n)_{\rm opt} \rightarrow (\varphi_{n-1}, \psi_{n-1})_{\rm opt} \rightarrow \ldots\ldots \rightarrow (\varphi_0, \psi_0)_{\rm opt}, \tag{6}$$

and the conformation (1) of the backbone of the peptide chain is determined. The procedure is illustrated in Fig. 6.

In the course of the procedure of determining the trajectory (6), the existence of the hydrogen or disulfide bonds has to be checked. When the condition is fulfilled, the sub-backbone is fixed, and the minimization is continued, but the optimal part of the sub-backbone remains unchanged.

4. The Process of Self-Replication

4.1. Self-Replication of Programs

There are two approaches to the program self-replication:

(a) making its own copy by a self-replicating program and passing over to the program the full autonomous control over its execution (von Neumann's concept, cf. (Neumann, 1966)).

(b) double formatting of program coding, which allows the program breakup in the self-replication process into two semantically equivalent programs and starts their execution as two independent programs, similarly as in the case of the cells of biological organisms.

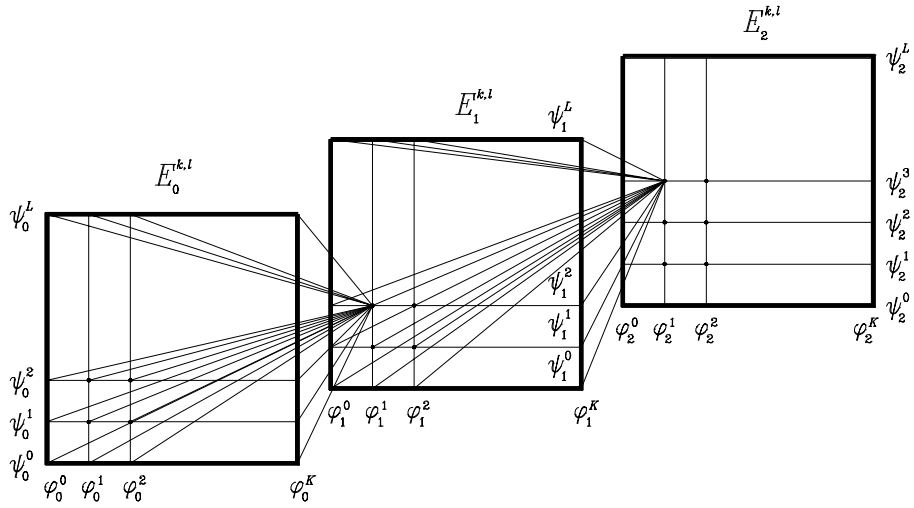


Fig. 6. Conformation determined by dynamic programming (Steps 0 to 2).

We will discuss these two cases in greater detail (Węgrzyn, 2001b).

Case (a). *Self-replication of a program through the generation of its own copy.*

The run of a self-replicating program in the case of making and activating its copy presented in Fig. 7 has the following form:

- the program ready for execution is placed into a selected set of programs,
- the management system is solicited to activate the program when it is placed in the selected set, based on and fulfilling the system rules,
- the activated self-copying program produces its own copy and forces its placing into the selected set mentioned earlier; as a result, the copy becomes an autonomous program and will be activated from the management system in a relevant time.

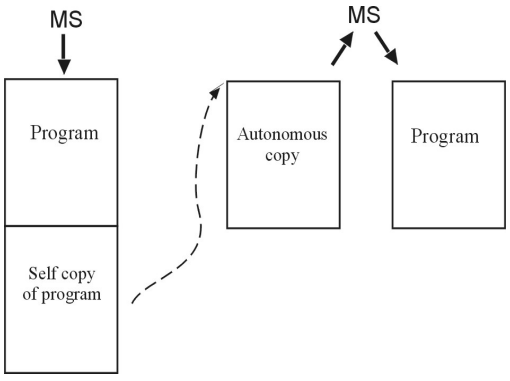


Fig. 7. Illustration of the self-replication process of a program through copying: (a) copy generation, (b) copy rendering, (MS – Management System).

Case (b). *Self-replication through the program division of the double recorded programs.*

The execution of a self-replicating program with a double record through its division is presented in Fig. 8.

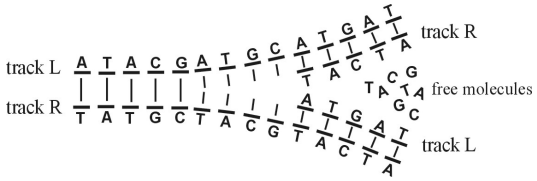


Fig. 8. Self-replication process for the program of DNA with a double format of coding.

The program consists of two semantically equivalent tracks L and R. The following relations exist between the corresponding nucleotides of the two tracks: A ↔ T and G ↔ C. The tracks break out at the moment of self-replication, and complement in accordance with the following principle: L is complemented with R, and R is complemented with L. As a result, the programs begin their work as two complete, independent programs.

4.2. Self-Replication of Objects Containing Internal Programs

The self-replication of objects containing internal programs takes place when an object has been constructed in some environment and, after finishing the program, it produces its own copies which subsequently start to build the consecutive objects.

The initiative cell which starts the construction of such an object has to posses not only a building program

exploiting the components of the surrounding environment, but also a program to elaborate successive initiative cells, which, on the other hand, are able to stimulate the development of the consecutive objects.

The programs of object building require large-scale packaging, and nanotechnologies and nanosystems of informatics are tools that may make a successful technical realization of connective self-replication, i.e. objects with internal software, possible.

Let us assume that there is a set of determined elements, or the environment. Also assume that the energy provided by the outside ensures the mobility of the elements and, as a result, free movement conditions are fulfilled in the environment area.

The object means a part of the environment selected in such a way that the exchange of elements between the environment and the object is possible provided that the determined conditions are fulfilled.

Consequently, the object can gather elements necessary for its development and can develop in accordance to its internal program. If, as a consequence of the object development, a partition of other objects which are able to continue the successive partition and descendant objects which are able to continue their independent development processes takes place, then such a phenomenon is called a self-replication process.

A run of a self-replicating process in the case of partition of an object into two descendant objects is presented in Fig. 9.

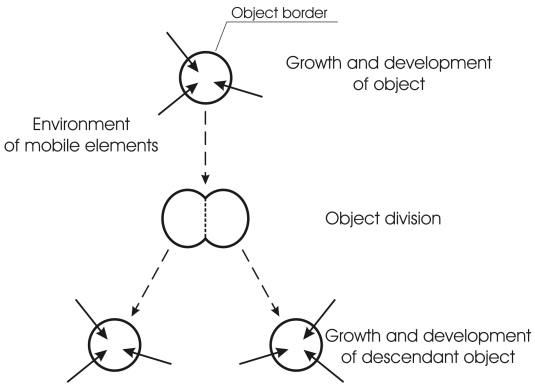


Fig. 9. Illustration of an object self-replication process.

4.3. Analysis of Division Processes

Let T_p be the time necessary for the preparation of an object to the division process. In addition, assume that this time is proportional to the number of completed objects appearing as a result of the partition process

$$T_p = k(n-1), \tag{7}$$

where k means the proportionality coefficient and n is the number of completed objects appearing as a result of the partition of one object.

Consider the problem of determining integer values n in the range $n \in [2, \infty]$ such that the number l of the objects generated after the time interval T, which is assumed to be a multiple of T_p, reaches the greatest value.

If we adopt the assumption (7), then the number l as a function of n can be expressed by the following relation:

$$l(n) = n^{T/Tp} = n^{T/(k(n-1))}. \tag{8}$$

For $n \in [2, \infty]$ it can be proved that the following inequality holds:

$$l(n) > l(n+1). \tag{9}$$

Inequality (9) can be expressed in the form

$$n^{1/(n-1)} > (n+1)^{1/n}. \tag{10}$$

The correctness of (9) can be proved based on the properties of the continuous function

$$y(x) = x^{f(x)}, \tag{11}$$

where

$$f(x) = \frac{1}{(x-1)}.$$

Substituting $x = e^{\ln x}$, we have

$$\begin{aligned} \frac{\mathrm{d}y}{\mathrm{d}x} &= e^{f(x)\ln x}\left(\frac{\mathrm{d}f(x)}{\mathrm{d}x}\ln x + \frac{1}{x}f(x)\right) \\ &= \frac{1}{x-1}e^{(1/(x-1))\ln x}\left(\frac{1}{x} - \frac{\ln x}{x-1}\right) \\ &= \frac{1}{x-1}x^{1/(x-1)}\left(\frac{1}{x} - \frac{\ln x}{x-1}\right). \end{aligned} \tag{12}$$

Equation (12) implies that for $n \in [2, \infty]$ we have $\mathrm{d}y/\mathrm{d}x < 0$, since $1/(x-1) > 0$, $e^{(1/(x-1))\ln x} > 0$, and

$$\frac{1}{x} < \frac{\ln x}{x-1}.$$

Therefore (9) is proved.

During the time interval T, for $n = 2$ the maximum number l of objects is created from one initial object, just as in the case of cells of living biological organisms.

5. Conclusions

The analysis of development processes of a biological cell together with its self-replication presented above implies the following conclusions:

1. Biological cells together with the programs contained in them and the execution processes of those

programs form a single unified nanotechnological entity.

2. The programs of cell development and self-replication form a single nanosystem of informatics in the cell.

Nanotechnologies and nanosystems of informatics can therefore be said to form a basis for biological cell self-replication. This should be taken into account when attempts are made to implement technological systems capable of manufacturing self-replicating products and objects.

Acknowledgment

This research was supported by the Polish State Committee for Scientific Research under grant no. 7T11C 01721.

References

Güntert P. and Wüthrich K. (2001): *Sampling of conformation space in torsion angle dynamics calculations*. — Comp. Phys. Comm., Vol. 138, No. 2, pp. 155–169.

Hovmoller S., Zhou T. and Ohlson T. (2002): *Conforations of amino acids in protein*. — Acta Crystal., Vol. D58, No. 5, pp. 768–776.

Levitt M., Hirshberg M., Sharon R. and Daggett V. (1995): *Potential energy function and parameters for simulations of the molecular dynamics of proteins and nucleic acids in solution*. — Comp. Phys. Comm., Vol. 91, Nos. 1–3, pp. 215–231.

Liebecq C. (Ed.) (1992): *The White Book. Biochemical Nomenclature and Related Documents, 2nd Ed., IUPAC-IUBMB Joint Commission*. — London: Portland Press.

von Neumann J. (1966): *Theory of Self Reproducing Automata*. — Urbana, IL: University of Illinois Press.

Ramachandran G.N. and Sasisekharan V. (1968): *Conformation of polypeptides and proteins*, In: Adv. Protein Chem., Vol. 23. – New York: Academic Press, pp. 506–517.

Węgrzyn S. (2001a): *Nanosystems of informatics*. — Int. J. Syst. Sci., Vol. 32, No. 12, pp. 1389–1397.

Węgrzyn S. (2001b): *Self-replications of informatic systems*. — Bull. Polish Acad. Sci., Tech. Sci., Vol. 49, No. 1, pp. 161–165.

Znamirowski L. (2002): *Network simulation of the post-translational modifications in organic nanostructures*. — Studia Inf., Vol. 23, No. 2A (48), pp. 23–34.

Znamirowski L. and Zukowska E. (2002): *Parallel nanomodeling of ribosomal translation processes*. — Proc. ICSEE'02 *Western Multi Conf.*, Vol. 34, No. 1, San Antonio, pp. 111–116.

Received: 14 February 2003
Revised: 27 June 2003