SOA Based Multi-Agent Approach for Biological Data Searching and Integration

Veska Gancheva

***Abstract*—** Major challenge in the analysis of biological data is to propose an integrated and modern access to the progressively increasing amounts of data in multiple formats, and efficient approaches for their processing. Models for extraction and integration of large amount of genomics data, as well as problems related to heterogeneity, distribution and compatibility of data are presented in this paper. SOA based multi-agent approach for biological data searching and integration is proposed. A conceptual architecture for integrating of distributed biological data based on SOA is designed. The architecture is aimed to automate the data integration and allows the rapid management of large volumes of diverse data sets represented in different formats - relational, NoSQL, flat files. The integration of different databases is solved by using multi-agent architecture. The integration system consists of services for transforming the common request into a specific language request for each local database, depending on its type. The conceptual database integration is solved by applying translating query approach. Each integrated database is represented by a separate conceptual scheme called a virtual scheme. This scheme is generated in the collating process, which compares structural elements from the database to the conceptual model. Service oriented multi-agent system for searching of biological data from different sources that sends queries to multiple databases and then compiles the results into a list, depending on the type of source is developed. The system allows the user to set search criteria and access multiple databases simultaneously. The services allow the system to be accessed over the Internet by multiple clients (mobile phones, web browsers, desktop applications) and serving a wide range of users simultaneously.

***Keywords*—**biological database, data integration, heterogeneous data, multi-agent system, SOA**.**

I. Introduction

WITH the development of bioinformatics, the number of biological data collected and the number of databases in which they are stored are continuously increasing [1]. One of the fundamental science fields, strongly dependent from the development of big data, is the molecular and computational biology field [2]. Databases are maintained by various organizations and institutions dealing with human genome research, virus testing and their mutations, protein research, drug synthesis, and so on. A major problem for the integration of biological data is the structure and format of the data. They are not always standardized and data access is not centralized, making it extremely difficult and time-consuming to search for results in all databases.

Most publicly accessible databases provide access to their data through the WEB. Generally, the result is in HTML format after the search criteria are filled in. Each database has a different access interface; it works with different data exchange protocols, which means that the same search query has to be translated for the corresponding database.

The work presented in this paper is a part of a project that offers a scientific platform for adaptive in silico knowledge data discovery based on big genomic data analytics. The focus is on advanced information technologies and the fourth scientific research paradigm Data Intensive Scientific Discovery (DISD) in support of precision medicine, specifically, for the case study of fighting breast cancer.

An innovative highly scalable and locality aware method for multiple nucleotide sequences alignment is designed [3]. The algorithm is iterative and is based on the concept of ABC metaheuristics and the concept of algorithmic and architectural spaces correlation. The metaphor of the ABC metaheuristics has been constructed and the functionalities of the agents have been defined. The conceptual parallel model of computations and the algorithmic framework have been designed. Parallelization and optimization of the multiple sequence alignment software MSA_BG utilizing MPI and OpenMP in order to improve the performance is proposed [4]. A high-performance environment integrating various services and middleware to facilitate access to grid resources for carrying out scientific experiments in the area of bioinformatics is proposed [5]. The environment is built up in order to enable parallel computer simulations on a grid infrastructure increasing the efficiency of the computations and allowing scientists easily and user friendly access.

The computational flow of in-silico knowledge data discovery has been presented and analyzed and the beneficial outcomes for the case study of genome mapping based on computer model of RNA revealed [6]. Conceptual model of big genomic data ecosystem has been suggested and the relevant most popular genomic data platforms revealed [7]. In silico knowledge data discovery pipeline for genome mapping based on promoters has been built up and the functionality of each stage of the pipeline has been defined.

This paper presents the outcomes of research project "Intelligent Method for Adaptive In-silico Knowledge Discovery and Decision Making Based on Analysis of Big Data Streams for Scientific Research", contract DN07/24, financed by the National Science Fund, Competition for Financial Support for Fundamental Research, Ministry of Education and Science, Bulgaria.

Veska Gancheva, PhD is Associated Professor at Technical University of Sofia, Department of Programming and Computer Technologies, Bulgaria; e-mail: vgan@tu-sofia.bg.

A platform for adaptive knowledge discovery and decision making based on big data analytics is proposed in [8]. The major advantage is the automatic generation of hypotheses and options for decisions, as verification and validation are performed using standard data sets and expertise of scientists. The tools for utilizing the platform are scalable framework and scientific portal to access the knowledge base and the software tools, as well as opportunities to share knowledge, and technology transfer. Web portal provides as services access and extraction of biological data and execution of program implementations for big genomics data analysis. An integrated approach for support of the knowledge discovery and decision making from big data analytics, based on adaptive machine learning and adaptive procedures for generating rules according to the goal of scientific research is explained.

A conceptual architecture for an integrated and modern access to the exponentially growing volume of data in multiple formats aimed to automate the integration of genomics databases is presented in this paper, which is a part of the scientific platform for adaptive in silico knowledge data discovery based on big genomic data analytics [8].

The paper is structured as follows: Biological data integration issues are discussed in Section II. The proposed multi-agent approach for data searching and integration is explained in Section III. Section IV is focused on the conclusions and future work.

II. Biological Data Integration Issues

The organization of biological data is a challenge that determines the storage and maintenance of the vast data sets:

- The volume of data has increased almost exponentially over the last decade.
- There are new types of data and thus develop new biological concepts.
- There is no standardization in the data nomenclature.
- Most often, data is stored in flat files and in relational databases: about 70% of the data is stored in a text or static image; the remaining 30% are data stored in different databases organized from indexed files to specialized relational databases.

The strong decentralization of biological data, the high degree of differences in terminology, the specificity of records, the presentation of data, and the format for querying data and information require the study of the possibilities of creating automated procedures for the databases integration. The goal is to achieve much more than simply retrieving and modifying data, because worldwide the dependence on access to data and information on practicing a profession is steadily increasing. This requires research to provide the necessary data and information resources that need to be comprehensive, easy to use, and linked to other databases or information. The heterogeneity and decentralization necessary to look for methods those provide access to current data. This requires the integration of large and different databases / information / knowledge related to the different levels of presentation.

A. Data Heterogeneity, Distribution, Changeability and Interoperability

Sequencing the human genome and the progress of research in the field of proteomics and molecular structure has led to the need for biological databases. Multiple biological databases have been developed worldwide for the storage of nucleotide and protein data - GenBank, EMBL, PIR, NCBI, Swiss-Prot, KEGG, GENES, and PDB. The availability of multiple databases that store heterogeneous information, in turn, solves issues related to heterogeneity, interoperability, complex data structures, and integration.

Biological knowledge is distributed in specialized databases/data sources. Each database has its own complex data structures reflecting the scientific concept of the model [9]. Many data sources have overlapping data elements with conflicting definitions. Data sources are not standard and often are not well documented. The integration and conversion of data from heterogeneous sources is very important for the effective use of biological information. It is important to interpret the different data formats, download data from different sources, and convert to integrate the information.

The data is obtained from different sources. They are in multiple formats (flat files, relational tables, text files, etc.) Biological data sources are characterized by an extremely high degree of heterogeneity in terms of the type of data model used, pattern of a given model, as well as incompatible formats and nomenclatures of the values [10]. In addition, a large number of unsustainable and inconsistent biological unit identifiers are often used as a data source.

The heterogeneity of the data consists of many forms and at many levels during data integration from multiple databases. Different types of data and formats are also forms of heterogeneity. The heterogeneity of the databases can be seen in the following aspects:

1. Heterogeneity of the names. In this case, different databases store the same values, but the names of the attributes are different in different databases. This type of heterogeneity can be solved by a syntactic query attribute transformation.

2. Heterogeneity of the relational structure. The composition of the attributes in a complex structure varies, but the stored values are identical. Heterogeneity can be solved by a syntactic relational query transformation.

3. Heterogeneity of the values. In this case, the way the values are presented is different in different databases. It may include transformations of type and value.

4. Semantic heterogeneity. This is the most difficult form of heterogeneity. In this case, data stored in different databases form different assumptions, for example about what they represent or how they are collected.

5. Heterogeneity of the data model. The transformations between the data models and the differences between them are relevant.

6. Heterogeneity at time. It is associated with the changes in the database structure, the representation of the attributes and the values themselves.

Various types of heterogeneity can be combined. These categories of heterogeneity can also be subdivided into subcategories. A separate task in the processing and analysis of large biological data sets is the processing of annotations. Each data set can contain different types of annotations that are often presented as semi-structured and / or structured data. By its nature, the information from such studies is described with a huge amount of structured and unstructured heterogeneous data, the presentation, processing and storage of which is still a serious challenge.

Biological databases are highly decentralized, with high levels of terminology, record specificity, data representation, and query formats [11]. This in turn is associated with problems with manually executing queries from multiple databases. Therefore, there is a need to automate the integration of biological databases, with much more than simply extracting and modifying the data. Records in different biological databases have different formats. Integration requires the use of mandatory formats in different databases, but large scale and redundancies make such integration impossible.

B. Data Integration Approaches

The iteration of data from different sources can be categorized by several criteria [12].

On the basis of the access model

Linking databases - consists of establishing links between the databases. In general, the links to the physical locations of other data sources are stored in one place. A major drawback of the approach is that it rather integrates multiple data sources without really integrating the data. Data sources remain with different data structures and formats.

Translating data - requires creation of a centralized data repository (centralized database) that stores data. Data from different sources is retrieved, if necessary transformed to a unified format, and recorded in the centralized repository. This allows execution of queries on the data without needing to be translated. The disadvantages of the approach may indicate that maintenance of actual data requires a lot of effort, a change in the structure of any of the databases integrated into the system could lead to a change in the structure of the system.

Translating query - this approach does not support centralized data repository. Instead, the searching query is subdivided into sub-queries for the corresponding database in the system.

Depending on the conceptual model

Pure mediation - for this purpose, mediators and agents are used to execute the queries. Mediators are responsible for providing all the information needed by the agent to return the result to the user. The system does not explicitly disclose the structure of the data accessed, making this approach less intuitive for users.

Single conceptual scheme - the approach uses a single data model for all databases in the system. The advantage is that the user can set request, taking into the single scheme of the system. Also, as with the data translation approach, changes to database schemas in the system can result in a change in the system schema.

Multiple conceptual scheme - this approach does not rely on a complete unification of the data structure. Instead, each database describes its own storage scheme and data model. The advantage of this approach is that changing the schema of one of the databases or removing it does not affect the system scheme. Client queries can be described by the use of code words from a particular area of ontology. Since individual databases can be part of different areas of different ontologies, some relevant results cannot be found when the query is executed.

Another approach that allows integration of multiple databases into a single system and does not fall into any of the above classifications is a comparison of schemes. In fact, it is used by the above approaches (except for linking databases where there is no real comparison of database schemas). This approach is a comparison of an equivalent record between two or more databases by a function of two-way transformation of one scheme to the other. The comparison of schemes is done mainly "manually", the increase in the schema of the respective database increases the difficulty of completing the comparison and increases the risk of errors. This requires the development of tools to automate the process.

Public databases are accessible through the WEB, and the result returned is in the form of HTML pages containing many service and non-user information. Genuine data have to be recognized, discovered and retrieved from the HTML page. For this purpose, it is best to use mediators and agents to retrieve the necessary information from the HTML page. Mediators embrace the HTML code by providing the necessary tools to the agent to retrieve valuable information. They are an HTML wrapper that implements standard date access interfaces. Mediators are built according to the structure of the HTML code for which they are created. Systems such as Ariadne and TSIMMIS describe the structure of their web pages using declarative languages and mediators are able to retrieve the necessary information based on knowledge of how is built the HTML page. In Lixto [13], web pages are described using XML configuration files describing the location of the actual information on the page. Very often, mediators are created for the specific web portal by a specialist who needs to know the scheme of the respective portal. Some systems such as RoadRunner [14] examine the similarities between several pages with the result of the portal in order to retrieve the scheme needed by the mediator.

OntoFusion [15] system integrates biological data using four XML configuration files: (1) a file containing database schema identification; (2) a file describing the way the requests are translated to the database; (3) a description of the HTML and (4) a description of the structure to retrieve the actual results. All these configuration files have be created by the system administrator and require a deep understanding of the web-based database that will be integrated.

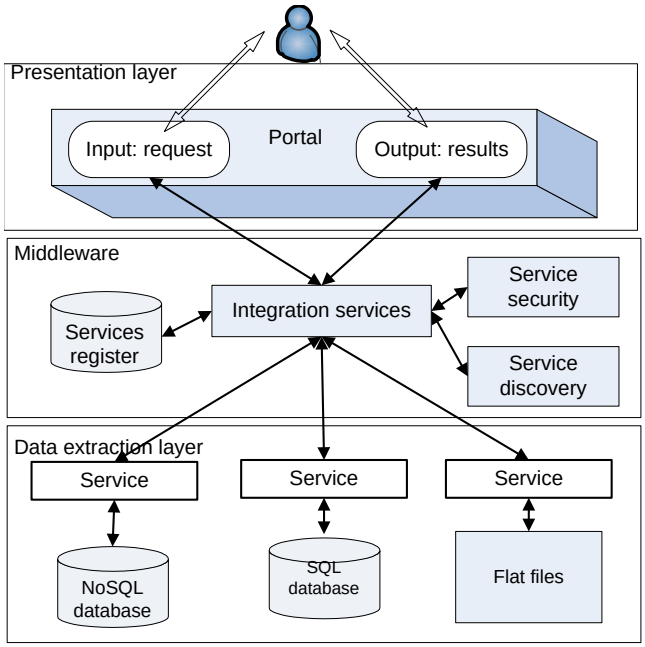

Fig. 1 Conceptual architecture for retrieval and integration of biological data based on SOA

III. Multi-Agent Approach for Data Searching and Integration

The requirements for scalable data integration systems for modern biology are indisputable due to the existence of very large, heterogeneous and complex datasets in the public database. Managing and merging these big data with local databases is a great challenge as it is the basis of computational analyzes and models that are then experimentally generated and validated through access portals for distributed modern high-performance infrastructure and software tools for big genomics data processing and visualization [8].

The aim is to propose a conceptual architecture for an integrated and modern access to the exponentially growing volume of data in multiple formats (Fig. 1).

The data integration service layer is the basic of the architecture and also the key to heterogeneous data integration. Metadata format vary greatly since which are grounded in heterogeneous sources. The different types of data are presented in XML format using rules for performing metadata operations. That first heterogeneous data is converted into unified XML format and then a basic layer services for data integration is created. The services are developed in an XML document based on the heterogeneous data sources.

The services achieve some data access: adding, modifying, querying, deleting and extracting the same functionality from the underlying service to the same service according to the data then forms the integrated data service function. Application layer call corresponding services according to the operational requirements, and then underlying data manipulation is specified by the service based on data parameters from the client calls; thus, when the underlying

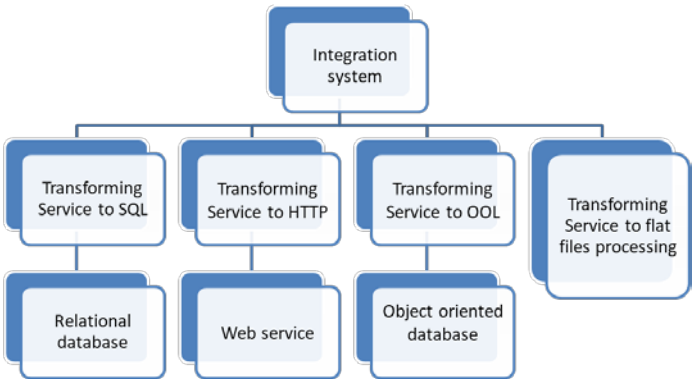

Fig. 2 Conceptual architecture of data retrieval and integration system

heterogeneous data source changes, the service is updated. Data integration implementation process is completely transparent to the user, which is compatible and interoperable in different systems.

The architecture allows the rapid management of large volumes of diverse data sets represented in different formats - relational, NoSQL, flat files. The integration system consists of services for transforming the common request into a specific language request for each local database, depending on its type (Fig. 2). Additionally, the possibility of making a permanent access to the state of research in order to compare the results with the available information (access to a constantly updated representation of all the accumulated knowledge in the relevant field) is further explored.

The integration of biological data from different databases is seen as being composed of two sub-problems: technological integration of different databases and conceptual database integration. The first problem is solved by using multi-agent architecture. Database agents act as mediators and are used to hide access procedures to the rest of the system and the access is only through mediators. The solution to the second problem is the need to overcome the heterogeneity of data at a semantic level. The translating query approach is used. In practice, each integrated database is represented by a separate conceptual scheme called a virtual scheme. This scheme is generated in the collating process, which compares structural elements from the database to the conceptual model.

Public databases can be integrated with virtual schemas. The problem with public databases is that they are not familiar with the database schema (tables, relationships, primary and secondary keys, etc.), and data from this type of database cannot be extracted with a relatively basic query. To create a virtual schema of a public web-based database, a database model and schema should be created.

Requests for web-based public databases are described through web interfaces and the corresponding URLs. Search criteria can be part of the URL or be part of the HTML access form. There is no unified way to set queries for this type of database. This requires describing the specification of the database in a configuration file that describes how to create queries, grouping, and others. It helps to translate the request for the respective web-based public database.

Once the query is created and executed, a result is returned in the form of HTML (intermediate page containing results links). Typically, this HTML contains links to query results. To retrieve the actual result from the list of links, it is necessary to describe how this information can be extracted, since it can be in the form of a table, a text file, etc.

A service oriented system for searching of biological data from different sources that sends queries to multiple databases and then compiles the results into a list, depending on the type of source is developed. The service allows the user to set search criteria and access multiple databases simultaneously. As a further feature, the service is realized as a multi-agent system, through the agent-oriented paradigm and appropriate agent-based technology.

The functionality is implemented by a set of services. The client accesses server logic through a web interface that contains the necessary search controls in multiple databases.

The services allow the app to be accessed over the Internet by multiple clients (mobile phones, web browsers, desktop applications) and serving a wide range of users simultaneously. Services delegate business logic to a layer of agents who perform the required tasks and then return an answer.

The user interface is built on Flash. The interface sends HTTP requests to the services and waits for a response. If a response returns results in a table, if it does not get an error is displayed. The Java programming language is chosen for the server, allowing efficient and productive software development, robust business and server logic. The Spring Framework is selected for service development.

The JADE environment is chosen to create agents business logic. JADE is an agent-oriented Java library, implementing the FIPA standard. It allows the creation of agents and the implementation of AOP. Jade4Spring is a Java library that allows agents to register with the IOC (Inversion of control container) on Spring. This allows easy creation and release of agents and avoids the need to manually initialize the JADE container. As a consequence, JADE acting agents are clearly indicated in an XML file, which improves and facilitates code maintenance.

For the serialization and deserialization of Java objects to the JSON format and vice versa, a Jackson library for JAVA was used. The library is used to transfer information between client and server.

The multilayer architecture design is shown in Fig. 3. The software application consists of several layers that communicate with each other.

- Web client - the part where the user enters the search data and selects the databases to be searched by the Jade agents, after which the searched data is sent to the web service via the communication channel.
- Client to server communication channel - connects the client to the server. Spring MVC technology, based on Java Servlets is used. Spring MVC facilitates the construction of such applications, taking care to manage multiple clients that can communicate with the service.

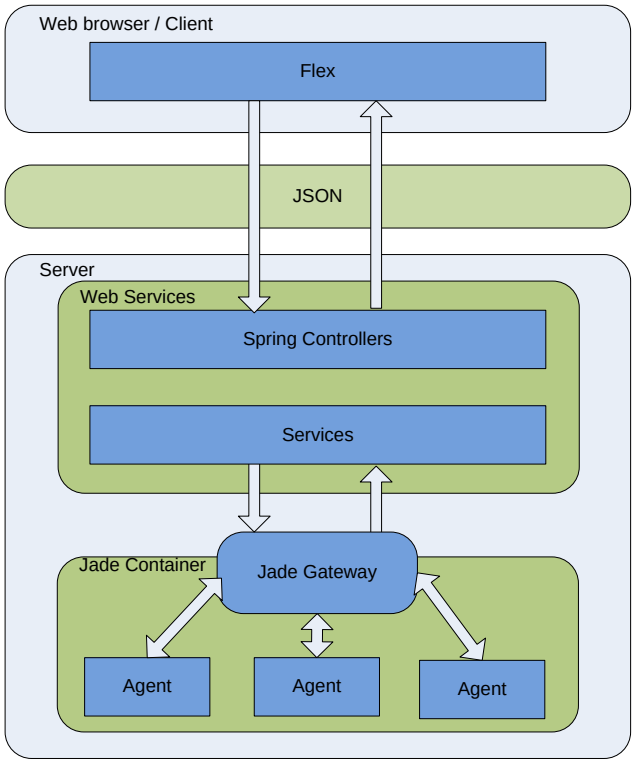

Fig. 3 Multilayer architecture of system for biological data searching

- Web services exposed via the REST interface - web services process requests received from the client and delegate them to business logic.
- Business logic implemented through agent-based technology - business logic performs the necessary manipulations to obtain a query result. It is done through JADE agents, which communicate with each other in order to successfully accomplish the task.

The communication between the client and the services is synchronous. This means that the client is waiting for a response from the server for a certain amount of time. If time expires, the client reports an error and updates their work.

The communication between the services and the agent container is also synchronous. Once the request has been processed by the service layer, a broadcast message is composed that is sent to the input of the JADE container (the Jade container input is also an agent). The thread that made the message falls asleep until it is awakened by an answer.

The agent's work and communication in the JADE container is asynchronous. Next to each "search" agent reaches message sent by “allocation tasks agent” (which implements JADE gateway interface) and it starts working. The “allocation tasks agent” receives asynchronous responses and decides when it is appropriate to wake up the sleeping thread from the service layer by sending an answer.

The created architecture makes it easier to add a new database type. All needed is to add an agent which knows how to work with the database. Weak connectivity between classes allows searching for heterogeneous data (databases, web search engines, registers) as communication with the services

is abstracted using a dictionary of answers. The dictionary contains symbol strings as values. This allows each agent to be coherent with the data format that works with and the only that is required is to serializes the response as a JSON object.

The scalability of architecture is determined by the fact that it is easy to place additional agents of the same type that adds a new level of parallelism to work. The jade4spring library has been used to manage the agents easily, and the application itself can be run as a server. Thus agents are described in Spring context as Spring beans. The appropriate configuration file is servlet-context.xml.

The purpose of each agent is to find as soon as possible the first results that are obtained in the corresponding database in which the search is to be performed. In order to achieve this, each agent is implemented as a separate class inheriting the SearchEngineAgent class. This is because at the moment all agents have a common function, namely fetchUrl, which is declared and implemented in SearchEngineAgent. The function is used to download the HTML from a submitted URL. Also, each agent implements the searchInEngine abstract function declared in SearchEngineAgent. Parameter of this function is string - data that is searched and returns string - serialized JSON object with found results. The function contains the actual code that, based on the HTML code with the results of the corresponding database, uses its own algorithm (different for each database) retrieves the results found and returns them as a JSON serialized object.

It is important to note that different search engines have different source HTML, that is, their specific strings that can locate individual search links are not only different, but may also be more than one, leading to the need for a completely different approach to finding them, and from there to a totally different algorithm. Therefore (due to the different search algorithms), it is right for each agent to search only in one database by implementing an algorithm in the abstract searchInEngine method, which is specific to it.

Delivering services over web addresses turns the application into a platform that can be used by other applications that need universal information searching.

IV. Conclusion

The paper presents a study of the models for storage and extraction of large volumes of biological data, the possibilities for integration of biological data as well as the problems related to heterogeneity, distribution, variability and interoperability of the data. A conceptual architecture for an integrated and modern access to the exponentially growing volume of data in multiple formats is proposed. The architecture allows the rapid management of large volumes of diverse data sets represented in different formats - relational, NoSQL, flat files. The integration system consists of services for transforming the common request into a specific language request for each local database, depending on its type.

The future work is to apply the proposed approach for biological data searching and integration on big genomic data analytics by practical experiments in the area of molecular biology for specific case study identifying regulatory genetic elements in sequenced genomes and in the area of medical genetics for specific case study knowledge discovery to predict the type and malignance of breast cancer through the available big data available by now concerning mutations in the genes associated with it, the level of expression and the associated epigenetic information. This will enable fast processing of clinical observations and laboratory analyzes data and comparison with the accumulated available data by now for the purpose of precise diagnosis and prediction.

References

[1] P. Chen, C. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data," *Journal of Information Sciences*, 275:314–347, DOI: 10.1016/j.ins.2014.01.015.

[2] A. Roy, "Trends in Computational Biology and Bioinformatics in the Era of Big Data Analytics," International Workshop on Bioinformatics in Fisheries and Aquaculture, ICAR-CIFRI, 2017, DOI: 10.13140/RG.2.2.21016.39680.

[3] P. Borovska, V. Gancheva, N. Landzhev, "Massively Parallel Algorithm for Multiple Biological Sequences Alignment," Proceedings of the IEEE International Conference on Telecommunications and Signal Processing (TSP), Rome, Italy, ISBN 978-1-4799-0402-0, pp. 638-642.

[4] P. Borovska, V. Gancheva, "Parallelization and Optimization of Multiple Biological Sequence Alignment Software Based on Social Behavior Model," *International Journal of Computers*, vol. 3, 2018, pp. 69-74, ISSN: 2367-8895.

[5] P. Borovska, V. Gancheva, N. Landzhev, "High Performance Grid Environment for Parallel Multiple Biological Sequence Alignment," Proceeding of ICCGI 2013: The Eighth International Multi-Conference on Computing in the Global Information Technology, ISBN: 978-1-61208-283-7, pp. 82-87.

[6] P. Borovska, "Big Data Analytics and Internet of medical Things Make Precision Medicine a Reality," *International Journal of Internet of Things and Web Services*, Volume 3, 2018, pp. 24-31, ISSN: 2367-9115.

[7] P. Borovska, "Big Data Analytics and Genetic Research,", International Conference on Big Data, Knowledge and Control Systems Engineering, BdKCSE'2017, 2017, Bulgaria.

[8] P. Borovska, V. Gancheva, "Platform for Adaptive Knowledge Discovery and Decision Making Based on Big Genomics Data Analytics," International Work-Conference on Bioinformatics and Biomedical Engineering IWBBIO 2019, accepted for publishing in Lecture Notes in Bioinformatics.

[9] C.. T. Yui, L. J. Liang, W. J. Soon, W. Husain, "A Survey on Data Integration in Bioinformatics," Springer-Verlag, Berlin Heidelberg, 2011.

[10] N. Paton, etc. (ed.) "Data Integration in the Life Sciences", 6th International Workshop, DILS 2009, Manchester, UK, July 20-22, 2009, Proceedings (Lecture Notes in Computer Science / Lecture Notes in Bioinformatics), Springer, ISBN-10: 3642028780, 2009.

[11] C. S. Rao, DVLN Somayajulu, H. Banka, S. Roy, "Feature Binding Technique for Integration of Biological Databases with Optimized Search and Retrieve," 2nd International Conference on Communication, Computing & Security [ICCCS-2012], pp.622- 629.

[12] W. Sujanski, "Heterogeneous database integration in biomedicine," *J Biomed Inform*, 34 (4) (2001), pp. 285-2986.

[13] R Baumgartner, S. Flesca, G. Gottlob, "Visual web information extraction with Lixto", In: Proceedings of 27th international conference on very large data bases (VLDB 2001), Rome, 2001. p 119–28.

[14] V. Crescenzi, G. Mecca, P. Merialdo, "RoadRunner: Towards automatic data extraction from large web sites", In: Proceedings of the 27th international conference on very large data bases (VLDB 2001), Rome, Italy, September 11–14; 2001. p 109–18.

[15] D. Perez-Rey, etc, "ONTOFUSION: ontology-based integration of genomic and clinical databases", Comput. Bio.l Med. 2006 Jul-Aug;36(7-8):712-30. Epub 2005 Sep 6.