

Crossing the interdisciplinary barrier: A baccalaureate computer science option in bioinformatics *

Travis Doom, *Member, IEEE*, Michael Raymer, Dan Krane, and
Oscar Garcia, *Life Fellow, IEEE*

Abstract – Bioinformatics is a new and rapidly evolving discipline that has emerged from the fields of experimental molecular biology and biochemistry, and from the the artificial intelligence, database, pattern recognition and algorithms disciplines of computer science. Largely because of the inherently interdisciplinary nature of bioinformatics research, academia has been slow to respond to strong industry and government demands for trained scientists to develop and apply novel bioinformatics techniques to the rapidly-growing, freely-available repositories of genetic and proteomic data. While some institutions are responding to this demand by establishing graduate programs in bioinformatics, the entrance barriers for these programs are high, largely due to the

* This work was supported by the National Science Foundation (NSF) under an Educational Innovation grant from the Computer and Information Science and Engineering (CISE) directorate, award #EIA-0122582.

significant amount of prerequisite knowledge in the disparate fields of biochemistry and computer science required for sophisticated new approaches to the analysis and interpretation of bioinformatics data. We present an undergraduate-level bioinformatics curriculum in computer science designed for the baccalaureate student. This program is designed to be easily tailored to the needs and resources of a variety of institutions.

Index Terms – Bioinformatics, curriculum, genomics, incorporation of research results in undergraduate education.

1 Introduction

Bioinformatics is a new and rapidly evolving discipline that has emerged from the fields of experimental molecular biology and biochemistry, and from the the artificial intelligence, database, pattern recognition and algorithms disciplines of computer science. Bioinformatics research explores the functional relationships between the composition of the *genes* within the context of the *genome* and the structure and function of the proteins encoded by these genes. Because the interaction of proteins largely determines metabolism, reproduction, form, and health, the implications of bioinformatics studies are far reaching. Recent advances in the experimental techniques of molecular biology have resulted in an explosive growth in the availability of molecular data. As a result, current bioinformatics research is often focused on the representation, analysis, annotation and mining of large databases of genome sequence information. In the future, the focus will shift to a functional analysis of the proteins produced by these genes and their interactions in the context of biochemical pathways. Bioinformatics techniques promise to provide information that: gets to the

heart of what life is; explains and improves the treatment of inherited and acquired diseases; aids in fighting famine and world hunger; and helps to produce a better environment.

Largely because of the inherently interdisciplinary nature of bioinformatics research, academia has been slow to respond to strong industry and government demands for trained scientists to develop and apply novel bioinformatics techniques to the rapidly-growing, freely-available repositories of genetic and proteomic data. While some institutions are responding to this demand by establishing graduate programs in bioinformatics, the entrance barriers for these programs are high, largely due to the significant amount of prerequisite knowledge in the disparate fields of biochemistry and computer science required to author sophisticated new approaches to the analysis of bioinformatics data.

There is a high demand for professionals with a background in bioinformatics. The sequencing and analysis of the human genome is one of the most complex computational problems currently being studied on a world-wide scale. Computer scientists are needed to analyze, index, represent, model, display, process, mine, and search large biological databases. This need is already extensive and will continue to grow. The genomic information available at the National Center for Biotechnology Information (NCBI) currently doubles every 14 months [1]. Industry analysts forecast that the market for genomic information alone (and the technology to use it) will reach an annual US \$2 billion by 2005 [2]. Both industry and the National Institute of General Medical Sciences (NIGMS) are having difficulty finding qualified individuals from other disciplines to investigate the kind of modeling and data analysis that researchers in the biological sciences now require [3].

Unfortunately, the educational opportunities available to undergraduate students wishing to partic-

ipate in this exciting enterprise are not yet sufficient to meet the anticipated demand [3–6]. Eric Lander (founder and director of Whitehead Institute Center for Genome Research) predicts that the current shortfall of bioinformaticians could be as much as fifty-fold [4]. The development of undergraduate opportunities in bioinformatics is essential to meeting future needs world-wide.

We have developed an undergraduate-level bioinformatics program that is unencumbered by the high entrance barriers associated with post-graduate bioinformatics education. We combine early training in the fundamental material necessary for a strong grasp of bioinformatics concepts and algorithms with junior- and senior-level bioinformatics research. The research component of our program is oriented towards application of existing bioinformatics methods to investigate current problems in molecular biology, and, for advanced students, development of novel techniques.

In Section 4.1 we present a model for a four-year baccalaureate degree program in computer science. Our goal in the development of this model is to provide exposure to the fundamental concepts required to produce engineers and scientists well prepared for postgraduate bioinformatics education and capable of carrying out research and development into new bioinformatic techniques and tools. Students will learn the algorithms, data representations, and ontologies at the core of current bioinformatics analyses. They will also learn how to implement these algorithms at the same time as they are exposed to the experimental techniques used by molecular biologists to gather data. By the end of their course of study, students will be ready to enter the bioinformatics job market or participate in on-going research projects involving analyses of molecular data.

2 Problem

Graduate programs in bioinformatics are beginning to emerge at universities world-wide [4]. Entrance requirements for such programs, however, require students with a specific prerequisite program of undergraduate study that is rarely made available as part of an organized program. Graduate bioinformatics programs must currently accept students with undergraduate degrees in either computer science or biology and have sequences of remedial or prerequisite courses designed to complement the knowledge already acquired by the students as undergraduates.

Students holding an undergraduate degree in computer science generally need to spend the majority of their first year of graduate study taking focused remedial courses in basic chemistry, biochemistry, molecular biology, and genetics. Students holding an undergraduate degree in biology generally spend the majority of their first year of graduate study in coursework covering introductory computer programming and data structures, entity-relationship modeling, databases, and artificial intelligence.

The second year of a graduate bioinformatics program is generally dominated by pre-existing graduate courses in computer science and biology. From computer science, courses in artificial intelligence, database, pattern recognition, and genetic algorithms are fundamental. From biology, a course sequence providing specialization in genetics, molecular biology, physiology, or ecology is considered highly advantageous. Finally, students from either background would require a course sequence covering contemporary algorithms and research techniques in bioinformatics. It is unlikely for this amount of material to be accommodated in a two-year course of study without significant preparation at the undergraduate level.

Due to the demanding entrance requirements, graduate programs alone may prove inadequate in providing the number of bioinformatics specialists that industry will require, partly because of the amount of the remedial coursework necessary for proper education. New undergraduate programs must be developed that incorporate a more specific (and shorter) biology sequence with a more focused computer science foundation. It may be necessary to redesignate some of the traditional core courses in some computer science programs (such as assembly language programming) as electives to allow students to increase their knowledge in the contemporary areas of IT knowledge applicable to bioinformatics (such as artificial intelligence, knowledge representation, pattern recognition, and data-mining).

It falls to four-year programs to provide opportunities and direction to students to meet the market demand for bioinformatics professionals and to better prepare students for entrance into graduate-level bioinformatics programs. As bioinformaticians must be equally versed in the languages of biology and computer science, this effort will require a fundamentally interdisciplinary approach. Furthermore, basic research in the field of bioinformatics is progressing rapidly; Professionals in fields such as bioinformatics and computational molecular biology must possess not only a strong grasp of computer science fundamentals, but must also be equally comfortable in the fundamentals of biology and biochemistry if they are to recognize and appreciate the results of their analyses.

2.1 Integration of computer science core material

Biology has become an increasingly data-driven science. Modern experimental techniques, including automated DNA sequencing, gene expression microarrays, and X-ray crystallography are

producing molecular data at a rate that has made traditional data analysis methods impractical. Bioinformatics methods are becoming an increasingly important aspect of the evaluation and analysis of experimental data in molecular biology. Computational modeling and prediction methods, such as comparative modeling of protein structure, are now reaching a level of sophistication that allows some experimentation to take place entirely within a computational framework. The July 2002 issue of *IEEE Computer* showcases the emergence of bioinformatics as a discipline in its own right [8].

Classically, computer science has focused on the study of computer hardware and software. A more contemporary view of information technology, however, must recognize that storage, transmission, and distribution of data make up a significant portion of the future demand on the discipline and on future computer professionals. This mandates a program of study emphasizing contemporary topics in databases and artificial intelligence.

From the discipline of computer science, a baccalaureate bioinformatics professional should have knowledge of: introductory programming, entity-relationship models, data structures, AI algorithms (search, optimization, list processing, pattern recognition, etc.), databases, formal and comparative languages (complexity, and specialized algorithm topics such as those explained in [9]). Additionally, a baccalaureate bioinformatics professional should have strength in at least one elective field from modeling and simulation, probability and statistics, visualization, pattern recognition, human-computer interaction (HCI), the development of complex bioinformatics systems (distributed systems), or evolutionary computation (EC). Mastery of all of these techniques is beyond the scope of a baccalaureate degree but are within the scope of post-graduate bioinformatics education.

2.2 Integration of biology core material

From the discipline of biology, a bioinformatics professional should have working knowledge of several life sciences fields, including genetics, environmental biology, physiology and biochemistry. Of these many possibilities, we propose to focus on the area of *molecular bioinformatics*. A professional in this field of study should understand genetics, molecular and cellular biology, chemical and physical aspects of the flow of genetic information from DNA to proteins, gene expression, replication, recombination and repair, and the experimental tools of molecular biology.

The amount of practical laboratory experience that should be possessed by an undergraduate bioinformatician is a point of debate. The results of DNA sequencing technology (and other *in vitro* and *in vivo* laboratory technologies) are published, annotated, and made available for analysis worldwide. The real problem is in extracting meaning from the glut of available data. Computationally generated results (*in silico* technologies) are becoming more prevalent in the field [2].

2.3 Incorporating research into the curriculum

Basic research in the field of bioinformatics is progressing rapidly. Students will be best served by a program of study that focuses primarily on the fundamental biological and algorithmic principles which give rise to bioinformatics techniques as a means of understanding and developing current and future analytical approaches.

For these reasons, *training in bioinformatics requires a strongly-integrated program of undergraduate and graduate student research activities that employs state-of-the-art algorithms and methods*

on biologically relevant data. The incorporation of substantive research activities into a bioinformatics tract for computer science students is both necessary and attainable for several reasons:

1. The discipline of bioinformatics is still in its infancy, thus many of the state-of-the-art solutions to important problems use direct approaches that are easily conveyed even to undergraduate students with limited background.
2. There exists a clear demarcation between research problems which utilize developed bioinformatics algorithms and the development of new algorithms. This facilitates a multidisciplinary approach to the understanding of bioinformatics methods.
3. World-wide collaborations in genome sequencing, protein-structure determination, and other areas have helped to bring about the sharing of bioinformatics research data and applications as a standard practice. Few other fields allow free access to largely unexplored research data and unperfected techniques as they are being developed. This affords invaluable opportunities for integrating the discovery of new knowledge into the core coursework of the program.

3 Methods

Bioinformatics professionals must be capable of communicating in both the languages of computer science and in the language of biology. Both disciplines are rich in technical terminology. The defining trait of a successful baccalaureate bioinformatician is not necessarily complete mastery of both fields, but rather a traditional mastery of one field and a comfortable familiarity with the other

[10].

The model we propose is designed to provide students with a traditional mastery of computer science through coursework available in existent accredited computer science programs. Additionally, we recommend coursework common in contemporary baccalaureate biology programs designed to provide students with opportunities to develop a “comfortable familiarity” with the language and concepts of biology crucial to bioinformatics. Finally, we recommend specialized bioinformatics training at an introductory level (during the sophomore year) and as a capstone (during the senior year) to provide students with opportunities to become familiar with the use and development of the tools of bioinformatics.

The introductory course, “Introduction to Bioinformatics”, is offered to sophomore-level students. This course, offered early in their program of study, presents the fundamental concepts of bioinformatics and provides a tools-oriented approach toward solving informatic problems. This course has minimal prerequisites and is designed for students with at least one course in computer science or biology. Student projects in the course are completed on a team basis, such that each team has at least one “expert” in computer science and one in biology. This collaboration fosters the ability to communicate concepts between the two disciplines – the hallmark of a trained bioinformatician. This introductory course is designed for students with little or no experience in bioinformatics. In-class lectures for this course focus on pen-and-paper implementation of algorithms, so the prerequisite experience in data structures and structured programming is minimized.

In contrast, the capstone course, “Algorithms for Bioinformatics”, is offered to students in their senior year. This course assumes that the incoming student is well-versed in both the fundamentals

of computer science and the fundamentals of biology and focuses on the application of contemporary algorithmic techniques, such as dynamic programming, to problems in bioinformatics.

The development and instruction of these two new courses are designed to be the only additional resources needed to implement a bioinformatics tract in existent computer science departments. It is critical, in our opinion, that these courses be co-taught by faculty from both the department of computer science and the department of biological sciences in order to provide the appropriate interdisciplinary mix. At Wright State University (Dayton, Ohio), these courses are co-listed as CS/BIO courses and open to students majoring in either discipline. The objectives of these two new courses are now detailed.

3.1 Introductory course: Introduction to bioinformatics

The goal of this 4 quarter credit hour (3 semester credit hour) introductory bioinformatics course is to present a tools-oriented approach to bioinformatics emphasizing data structure in DNA, data searches, pairwise alignments, substitution patterns, protein structure prediction and modeling, proteomics, and the use of web-based bioinformatics tools. This course also introduces students to beginning programming skills in Perl, the most common computer language for biological data analysis. The lectures focus on common classes of problems in bioinformatics, and in-class solutions are implemented using a language-independent, pencil-and-paper approach. Course objectives include development of a solid understanding of Perl basics, familiarity with existing computational approaches to solving problems in bioinformatics, and the skills necessary to continue towards advanced bioinformatics training.

- Module One

- (From Biology) Major biological issues that define the discipline: information storage in DNA, protein structure and function, the tools of molecular biology.
- (From Computer Science) Programming environment: command line Unix, redirection, pipelines for STDIN and STDOUT, introduction to the Perl programming language.
- (Lab) DNA isolation and gel electrophoresis.

- Module Two

- (From Biology) Simple pairwise alignments: simple searches, gaps, and techniques for scoring, strategies for efficient searches.
- (From Computer Science) Programming primitives: Scalar and array variables in Perl, basic I/O, program control.
- (Lab) DNA sequencing demonstration and counting nucleotides.

- Module Three

- (From Biology) Substitution patterns and phylogenetics: causes of variation in genes and lineages, molecular phylogenetics and phylogenetic trees.
- (From Computer Science) Programming decomposition: functions and subroutines, parameter passing.
- (Lab) PCR demonstration and finding open reading frames.

- Module Four

- (From Biology) Statistical and parsimony-based approaches to phylogenetics: distance matrix methods, cluster analysis and multiple sequence alignments, inferred ancestral sequences, molecular phylogenies.
 - (From Computer Science) Programming tools for bioinformatics: string manipulation and pattern matching in Perl, strategies for efficient searches: exhaustive, heuristic, and branch and bound.
 - (Lab) Sterile techniques demonstration and parsing molecular data repositories.
- Module Five
 - (From Biology) Gene recognition: gene structure and density, introns and exons, transposition and repetitive elements, introduction to gene microarrays.
 - (From Computer Science) Proteomics: predicting RNA secondary structure, tools for molecular visualization and structural modeling, tools for ligand screening, inhibition, and drug design.
 - (Lab) Final research project.

3.2 Capstone course: Algorithms for bioinformatics

The goal of this 4 quarter credit hour (3 semester credit hour) capstone course in bioinformatics is to provide a theory-oriented approach to the application of contemporary algorithms to bioinformatics. Graph theory, complexity theory, dynamic programming and optimization techniques will be introduced in the context of solving specific computational problems in molecular genetics.

- Module One

- (From Biology) Review of basic concepts of molecular biology: mechanisms of molecular genetics, transcription, translation, protein synthesis, and how the genome is studied.
- (From Computer Science) Review of data structures and complexity: elementary data structures and basic graph terminology, introduction to interval graphs, review of analyzing and designing algorithms, recurrence relations, polynomial and non-polynomial growth, introduction to the concept of NP-completeness, decision, and optimization problems.
- (Lab) Graph structures.

- Module Two

- (From Computer Science) Introduction to dynamic programming: elements of dynamic programming, optimal string alignment using dynamic programming, double dynamic programming.
- (From Biology) Sequence comparison and searching: global, local, and semi-local comparison, general and affine gap penalty functions, comparing multiple sequences, BLAST and FASTA.
- (Lab) Sequence comparison.

- Module Three

- (From Biology) Fragment assembly of DNA: base call errors, unknown orientation, repeated regions, and other biological complications.

- (From Computer Science) Optimization algorithms: elements of greedy strategy, greedy and heuristic algorithms for shortest common super-string, reconstruction, and multi-contig
- (Lab) Fragment assembly.

- Module Four
 - (From Biology) Physical mapping of DNA: restriction site mapping, hybridization mapping, and mathematical models for mapping strategies.
 - (From Computer Science) Molecular structure prediction: recurrence relations for determining total free energy of a structure, protein folding, branch and bound techniques for protein threading.
 - (Lab) Protein threading.

- Module Five
 - Contemporary algorithms for bioinformatics research: Introduction to the use of classifiers, genetic algorithms, Markov models, and other contemporary tools for research computation.
 - (Lab) Final research project.

The material presented in these two courses serves as a unifying element for a bioinformatics program otherwise consisting of existing coursework in computer science and biological science. Although designed in the framework of a computer science degree with a bioinformatics focus,

these courses can also be used to provide students majoring in biological sciences with similar bioinformatics opportunities.

To maximize student learning, we teach these courses using an active/cooperative learning approach [11–13]. Given the intensity of the course, it is vital that students are engaged in activities other than simply listening to lectures and taking notes. Students should be involved in talking and listening to one another, or writing, reading and reflecting individually throughout both courses. Integration of these courses into existing curricula is substantially aided by the creation of interdisciplinary project teams in which students with a stronger biology background take the lead in experimental design and data interpretation while students with a stronger computer science background take the lead in the development and implementation of representation methodologies and optimized solution-finding algorithms.

Additionally, as part of the capstone, student teams are asked to complete a formative project with research application. Students are afforded the opportunity to apply their own expertise on a project of their choosing or provided an opportunity to work on one of the several projects underway in the Bioinformatics Research Group at Wright State University.

4 Implementation

This model program was accepted as an official option in the computer science degree program at Wright State University in Spring 2002. Prior to this adoption, undergraduate students wishing to study bioinformatics worked with their advisors to develop individual programs of study which

required successful petition for graduation. We concurrently implemented a related bioinformatics degree program for the department of Biological Science.

Although the two related four-year courses of study differ significantly in their upper-division coursework, they were designed so that requirements are satisfied in tandem during their first two years in order to allow students to defer specialization in Computer Science or Biology until their junior year. During the first two years of study, both curriculums include a two-year sequence of computer science coursework (including programming and data structures), a two-year sequence of biology (including molecular genetics and cell biology), a two-year chemistry sequence (including organic chemistry), and a two-year sequence in mathematics (including calculus, discrete mathematics and statistics). Upper-division coursework in the related programs differs but retain major areas of commonality, including operating systems, artificial intelligence, and database design, advanced topics in biochemistry and molecular cell biology, and the capstone course in bioinformatics developed for this option.

4.1 A model computer science curriculum for undergraduate bioinformatics

We now present a curriculum proposal for computer science which we believe is in accordance with CAC/CSAB/ABET standards [14], yet incorporates specific sequences in chemistry and biology with a computer science foundation focused towards the development of contemporary IT skills critical to bioinformatics. In order to meet our objectives, it was necessary to remove several traditional topics from the classical computer science curriculum for this option. We have removed

only those topics which, according to our interpretation of accreditation, and not essential to the student of computer science and thus may have to play an elective role in the education of interdisciplinary bioinformaticians. Knowledge of calculus-based physics, for instance, is not as important for students preparing for careers in bioinformatics as it is for those interested in digital signal processing. Many of the traditional focuses of computer science that are not required by our program are, however, accepted as elective coursework.

To facilitate the implementation of this program, we have introduced only two new courses (presented in Section 3). Both courses are co-taught by faculty from both the Department of Computer Science and the Department of Biology. This first course, *Introduction to Bioinformatics*, is designed not only for students in the bioinformatics program, but as an elective for all biology or computer science students who wish some exposure to the field. Thus, this sophomore-level course has a relatively wide appeal and its enrollment exceeds the number of students entering the bioinformatics program proper.

We believe that this model program in bioinformatics can be successfully adopted by most existing computer science departments. The additional resources required for this program are small – only two new courses are introduced as regular offerings. Furthermore, we recommend that the courses be co-taught with both computer science and biological sciences faculty, thus sharing the additional overhead costs between two academic units.

Computer Science - Bachelor of Science

Model option in bioinformatics

Total Quarter Credit Hours: 195

I Humanities and General Education (48)

- English (8)
- Humanities (34)
- Social Implications of Computing (3)
- Technical Communications (3)

II Computer Science Requirements (60)

Core (32)

- One year programming sequence (12)
- Intro. to Comp. Information Sys. (4)
- Digital Computer Hardware (4)
- Computer Organization (4)
- Data Structures and Software Design (4)
- Intro. to Bioinformatics (4)

Advanced (28)

- Database Systems (4)
- Artificial Intelligence (4)
- Operating Systems (4)
- Comparative Languages (4)
- Algorithms for Bioinformatics (4)

- Electives (8)

III Math and Science (87)

Biology (29 hours)

- One-year biology sequence for science majors (12)
- Molecular Biology (4)
- Molecular Genetics (4)
- Cell Biology (4)
- Electives (5)

Chemistry (33 hours)

- One-year inorganic chemistry sequence (15)
- One-year organic chemistry sequence (18)

Mathematics (25)

- One-year calculus sequence (15)
- Elementary Matrix Algebra (3)
- Discrete Mathematics (3)
- Statistics (4)

It is expected that this model bioinformatics program can be reasonably modified to meet CAC/CSAB/ABET requirements for a bachelor of science degree in computer science at most universities. The model

program is designed with a total of 195 credit hours and can be completed in a four-year program of study. The general education requirement of 48 quarter credit hours (CAC requires 45) includes coursework designed to develop and apply written communication skills (English, Technical Communications), oral communication skills (Technical Communications), and coverage of social and ethical implications in computer science (Social Implications of Computing).

As shown in the model program, there are a total of 60 quarter credit hours of formal computer science coursework (CAC requires 60), including 32 quarter credit hours of core computer science coursework (CAC requires 24) and 28 quarter credit hours of advanced coursework (CAC requires 24). The introductory computer science sequence meets the CAC guidelines of building proficiency in high-level programming and software design. CAC guidelines are also met by including courses focusing on data structures and entity-relationship models (Intro. to Information Sys., Data Structures), computer organization, algorithms (Data Structures, Algorithms for Bioinformatics, and exposure to a variety of programming languages and systems (Comparative Languages, Operating Systems)

The 29 quarter hours of Biology coursework and the 33 hours of Chemistry coursework (CAC requires only 18 hours of science coursework) include the CAC required one-year lab sequence for science or engineering majors. The 25 quarter hours of mathematics in the model (CAC requires 23) includes all CAC mandatory mathematics topics, including discrete math, differential and integral calculus, and probability and statistics. Together, the model program contains a total of 87 quarter credits hours of math and science (CAC requires 45). The biology electives are provided to afford students the opportunity to obtain specialized knowledge in upper-division Biology courses and to simplify obtaining a minor in Biological Sciences for students desiring such a cognate.

The credit hours for electives in biology can be easily be transfered into other program areas for programs that require such flexibility without violating CAC guidelines.

5 Conclusion

Computer science is a path to understanding genomes just as biology helps us in understanding living organisms. It is hard to imagine a more significant area where we must hone our methods of questioning than bioinformatics. The competitive pressure and rewards for progress in bioinformatics are high, and students can use them to prepare themselves to join this sought-after work-force. The creation of an undergraduate bioinformatics option in computer science and engineering is of utmost importance for global health, economic development, and the success of our students.

The central argument that we present for an undergraduate bioinformatics option within a Computer Science BS degree can be summarized as follows: (1) Students holding undergraduate degrees in Computer Science or Biology are generally required to remediate coursework from the other discipline if accepted to a postgraduate bioinformatics program. The number and chain of prerequisites that must be satisfied in either case requires about two years of coursework because course dependencies are such that they cannot be taken in parallel. (2) An assumption of two years of remedial coursework, in addition to the two years to obtain the MS degree, implies that it could take eight years of preparation for a student to obtain an MS degree in bioinformatics. (3) The alternative that we propose would lead to a BS degree in bioinformatics four years and an MS degree in bioinformatics in the standard six year time frame.

We believe that complementing existing degree options in computer science with the appropriate existing chemistry and biology coursework is the most realistic way to implement programs in bioinformatics. The model we present requires relatively few additional resources: two additional courses offered as infrequently as once every other year may suffice for small programs. Thus, this model not only meets the needs of research universities (such as Wright State University), but also provides potential direction for the many small, primarily liberal arts, colleges interested in providing bioinformatics education [7].

The model that we present is just that – a model. The content decisions for this model are based upon the contemporary needs of the bioinformatics industry and towards providing the academic background required for meeting the entrance requirements of postgraduate programs in bioinformatics. As is the case with computer science, it is impossible to include all potential fields of interest within the scope of a baccalorate degree. Each institution will need to address the specific implementation appropriate to their institution. The integration of bioinformatics resources, such as the two courses that we propose, with those resources already present within the computer science and biology programs at existing institutions will require the participation of faculty from both biology and computer science. We believe that this model will serve as a framework for initial dialogues and will help guide faculty towards the rapid development of bioinformatics programs that capitalize on the strengths of their specific institution.

References

- [1] National Center for Biotechnology Information (NCBI). <http://www.ncbi.nlm.nih.gov>-

/Database/index.html, Aug. 1 2002.

- [2] S. K. Moore, "Understanding the human genome," *IEEE Spectrum*, vol. 37, no. 11, pp. 33 – 35, Nov. 2000.
- [3] C. M. Henry, "The hottest job in town," *Chemical and Engineering News*, vol. 79, no. 1, pp. 47 – 55, Jan. 2001.
- [4] B. Schachter, "Bioinformatics moves to the head of the class," *Bio-IT World*, pp. 62 – 67, Jun. 2002.
- [5] T. Doom, M. Raymer, D. Krane, and O. Garcia, "A proposed undergraduate bioinformatics curriculum for computer scientists," in *Proceedings of the 2002 ACM Special Interest Group on Computer Science Education (SIGCSE 2002)*, Convington, KY, Mar. 2002.
- [6] T. E. Doom and O. N. Garcia, "Bioinformatics: An option in computer science," in *2001 Midwest Artificial Intelligence and Cognitive Science (MAICS) Conference*, Miami, OH, Mar. 2001.
- [7] B. Dyer and M. LeBlanc, "NSF workshop: Incorporating genomics research into the undergraduate curricula." <http://genomics.wheatoncollege.edu/>, NSF DUE-0126643, Wheaton College, Norton MA, June 2002.
- [8] "About this issue," *IEEE Computer*, vol. 35, no. 7, p. 3, Jul. 2002.
- [9] P. Baldi and S. Brunak, *Bioinformatics: The machine learning approach*. Cambridge: MIT Press, 1998.

- [10] D. Krane and M. Raymer, *Fundamental concepts of bioinformatics*. San Francisco: Benjamin Cummings, 2002.
- [11] D. Johnson, R. Johnson, and K. Smith, *Active learning: Cooperation in the college classroom*. Edina: Interaction Book Co., 1998.
- [12] D. Johnson, R. Johnson, and K. Smith, “Maximizing instruction through cooperative learning,” *American Society for Engineering Education (ASEE) Prism*, vol. 7, pp. 24 – 29, Feb. 1998.
- [13] B. Millis and P. Cottell, *Cooperative learning for higher education faculty*. Westport: Oryx Press, 1998.
- [14] Computing Sciences Accreditation Commission, “Criteria for accrediting programs in computer science in the United States.” <http://www.csab.org>, Aug. 1 2000.