TOPICS IN INTELLIGENT COMPUTING/TOPICS IN SOFT COMPUTING:
FROM QUANTUM COMPUTING TO COMPUTERS OF GENERATION OMEGA
Syllabus for the course CS 5354/4365, Fall 2020

CLASS TIME: MW1:30-2:50 pm, Room CCSB 1.0702

INSTRUCTOR: Vladik Kreinovich, email vladik@utep.edu, office CCSB 3.0404, office phone (915) 747-6951.

- The instructor's office hours are Mondays and Wednesdays 10:30-11:30 am, 12:30-1:30 pm, or by appointment.
- If you want to come during the scheduled office hours, there is no need to schedule an appointment.
- If you cannot come during the instructor's scheduled office hours, please schedule an appointment in the following way:
  - use the instructor's appointments page http://www.cs.utep.edu/vladik/appointments.html to find the time when the instructor is not busy (i.e., when he has no other appointments), and
  - send him an email, to vladik@utep.edu, indicating the day and time that you would like to meet.

  He will then send a reply email, usually confirming that he is available at this time, and he will place the meeting with you on his schedule.

MAIN OBJECTIVE: to learn advanced computing techniques such as quantum computing, and to learn how these techniques can be used in intelligent computing.

CONTENTS

We need faster computers. While modern computers are much faster than in the past, there are still many practical problems for which they are too slow. For example, it is, in principle, possible to predict with high probability where a tornado will go in the next 15 minutes, but even on modern high performance computers, this computation will require several hours -- too late for this result to be useful.

What physical processes can we use to speed up computations. Since we have been unable to achieve a drastic speedup by using the traditionally used physical processes, a natural idea is to analyze whether using other physical processes can help. This analysis is the main topic of this class.

How can we find physical processes that can help to speed up computations? A natural idea is that we need to find processes whose future behavior is computationally difficult to predict.

Indeed, if this behavior was not difficult to predict, then we would be able to replace the use of these processes with the corresponding computations at about the same time -- and thus, we would get a traditional computer that uses almost the same computation time, while what we wanted was a drastic increase in computational speed.

So, to decide which physical processes are appropriate for computation speed-up, we need to analyze the computational complexity of different physical phenomena -- to be more precise, complexity of the models describing different physical phenomena.

This leads to computational complexity: the 1st topic of this class. To be able to perform computational complexity analysis of different physical phenomena, we will first recall the main definitions of computational complexity -- worst-case time complexity, average time complexity, feasible algorithms, P and NP, and NP-hard problems.

Once we refresh on this background, we will start analyzing different physical phenomena from the viewpoint of their computational complexity.

Types of physical processes. Depending on what we know -- and what we can determine -- we can divide physical processes into three main types:

- processes for which we know the models that predict the results,
- processes for which the results are partly unpredictable, but for which we can predict some characteristics -- e.g., probabilities of different outcomes, and
- completely "lawless" processes, for which any predicting model -- even a model for predicting probabilities -- will eventually turn out to be wrong.

Let us analyze if and how processes of each type can be used to speed up computations.

Processes for which we know the models that predict the results. Most such processes are described by partial differential equations, in which the time derivative of all the quantities x(t) depends on their current values.

Usually, in such models, the dependence of the time derivative v(t) on the current values is computationally feasible. So, to predict the value x(t + h) for small h > 0, we can simply compute a computationally feasible expression x(t) + h * v(t). Thus, such processes cannot lead to a drastic computational speedup.

Processes for which the results are partly unpredictable, but for which we can predict some characteristics -- e.g., probabilities of different outcomes: main example. In modern physics, there are such process -- e.g., radioactive decay. These processes are described by quantum mechanics.

In quantum mechanics, in addition to differential equations that describe a *smooth* change in the system's state, we also have *abrupt* -- and *probabilistic* -- changes corresponding to measurements. And measurements are ubiquitous, since they are the only way by which we can gain some information about the quantum system.

Quantum processes can indeed speed up computations. For quantum systems, prediction indeed turns out to be NP-hard; see, e.g., Kreinovich et al. 1991. Not surprisingly, several schemes have been discovered for using quantum processes to speed up computations.

Quantum computing can help in solving all practical problems. From the general viewpoint, these schemes cover all possible applications of computers. Indeed, from this general viewpoint, what do we want?

- We want to understand how the world works, predict what will happen -- this is, crudely speaking, what science is about. For example, we want to understand where the tornado will turn.
- We also want to understand how can we improve the situation -- this is, crudely speaking, what engineering is about. For example, how can be make tornadoes change their course? How can we make houses less vulnerable to tornadoes?

- Finally, we want to communicate -- or not -- with others, so we need to develop techniques for communication only with the intended folks.

Quantum computing is useful in solving the main problems of science and engineering.

- In the general prediction problem, we need to find a model that fits all the observations.
- In a usual engineering problem, we need to find a design and/or a control that satisfies a given specification.

In most of these problems, once we have a model, a design, or a control, it is computationally feasible to check whether this model, design, etc. satisfies the given specifications, it is searching for a satisfactory model, design, etc. which is computationally intensive. (In other words, these problems belong to the class NP.)

To speed up such problem, we can use Grover's quantum search algorithm.

Quantum computing and Grover's algorithm: the 2nd topic of this class. In class, we will review the basic ideas of quantum computing, and explain the main ideas behind Grover's algorithm.

Need for optimization. In some cases, we do not just want to find not just a model, a design, or a control, but rather the *best* model, design, and control.

Quantum computing can help with optimization. It turns out that Grover's algorithm can speed up the solution of optimization problems as well; see, e.g., (Ayub et al. 2020).

3rd topic. Quantum optimization will be the 3rd topic of this class.

Quantum computing and communications. Due to its efficiency, quantum computing can break down the existing encryption algorithms such as RSA. Good news is that by using quantum effects, we can develop an unbreakable quantum crypography scheme; see, e.g., (Galindo et al. 2018).

4th topic. RSA algorithm, its quantum-related vulnerablity, and quantum cryptography will be the 4th topic of this class.

Randomness in general. Intuitively, a random sequence is a squence that cannot be easily computed. This idea leads to a formal definition of randomness via Kolmogorov complexity and Algorithmic Information Theory. Not surprisingly, the corresponding notions are difficult to compute -- e.g., Kolmogorov complexity is not algorithmically computable. Since, according to modern physics, random processes do occur in real life, the use of random processes may lead to yet another way to speed up computations; see, e.g., (Kosheleva and Kreinovich 2020).

5th topic. Kolmogorov complexity, randomness, and computablity of the corresponding notions will be the 5th topic of this class.

Completely "lawless" processes. Many physicists believe that no matter how complex theories we propose, there will always be some new phenomena that would require us to modify these theories. In computational terms, this means that the sequence of observations is not computable. Not surprisingly, this idea leads to the possibility of speeding up computations; see, e.g., (Kosheleva et al. 2014).

6th topic. Study of such "lawless" sequences will form a (relatively short) 6th topic of this class.

Processes about which we do not know much but that show promise. Another possibility is to look for processes which are promising -- in the sense that they are suprisingly faster than they should be. A biological example of such a process is given in (Kosheleva and Kreinovich 2020a).

7th topic. This will be an even shorter 7th topic of this class.

Another possibility: using physical processes with unusual space and time. Up to now, we considered processes that can potentially speed up computations within the usual concepts of physical space and physical time. However, many physical theories -- e.g., special and general relativity theories -- are based on changing the usual concepts of space and time. Many of these changes can lead to speed up of computations:

- we can use the fact that, according to relativity theory, time slows down for fast particles or in the presence of a strong gravitational field, for example, near the black hole; see, e.g., (Kosheleva and Kreinovich to appear);
- we can use the fact that in curved space-time, volume changes, so we may be able to fit more processors working in parallel and thus, speed up computations; see, e.g., (Morgenstein and Kreinovich 1995);
- we can use possible acausal processes; see, e.g., (Koshelev and Kreinovich 1997); and
- we can use models in which space and time are discrete; discrete computations are usually more difficult than continuous ones, so if we have a real-life discrete system, this can potentially speed up computations; see, e.g., (Alvarez et al. 2019).

8th topic. Studying how different space-time models can speed up computations will be the last topic of this class

MATERIALS USED IN THE CLASS: there is no textbook; instead, we will use the following papers; they are listed in alphabetic order:

R. Alvarez, N. Sims, C. Servin, M. Ceberio, and V. Kreinovich, "If Space-Time Is Discrete, It May Be Possible to Solve NP-Complete Problems in Polynomial Time", University of Texas at El Paso, Department of Computer Science, Technical Report UTEP-CS-19-85b, 2019.
file in pdf

C. Ayub, M. Ceberio, and V. Kreinovich, "How quantum computing can help with (continuous) optimization", In: M. Ceberio and V. Kreinovich (eds.), "Decision Making under Constraints", Springer Verlag, Cham, Switzerland, 2020, pp. 7-14.
file in pdf

O. Galindo, V. Kreinovich, and O. Kosheleva, "Current quantum cryptography algorithm is optimal: a proof", Proceedings of the IEEE Symposium on Computational Intelligence for Engineering Solutions CIES'2018, Bengaluru, India, November 18-21, 2018.
file in pdf

M. Koshelev and V. Kreinovich, "Towards Computers of Generation Omega - Non-Equilibrium Thermodynamics, Granularity, and Acausal Processes: A Brief Survey", Proceedings of the International Conference on Intelligent Systems and Semiotics (ISAS'97), National Institute of Standards and Technology Publ., Gaithersburg, MD, 1997, pp. 383-388.
file in pdf

O. Kosheleva and V. Kreinovich, "Physical Randomness Can Help in Computations", University of Texas at El Paso, Department of Computer Science, Technical Report UTEP-CS-20-02, 2020.
file in pdf

O. Kosheleva and V. Kreinovich, "A Mystery of Human Biological Development -- Can It Be Used to Speed up Computations?", University of Texas at El Paso, Department of Computer Science, Technical Report UTEP-CS-20-04, 2020a.
file in pdf

O. Kosheleva and V. Kreinovich, "Relativistic effects can be used to achieve a universal square-root (or even faster) computation speedup", In: A. Blass, P. Cegielsky, N. Dershowitz, M. Droste, and B. Finkbeiner (eds.), Fields of Logic and Computation III, Springer, to appear.
file in pdf

O. Kosheleva, M. Zakharevich, and V. Kreinovich, "If many physicists are right and no physical theory is perfect, then by using physical observations, we can feasibly solve almost all instances of each np-complete problem", Mathematical Structures and Modeling, 2014, Vol. 31, pp. 4-17.
file in pdf

V. Kreinovich, A. Vazquez, and O. M. Kosheleva. "Prediction problem in quantum mechanics is intractable (NP-hard)," International Journal of Theoretical Physics, 1991, Vol. 30, No. 2, pp. 113-122.

D. Morgenstein and V. Kreinovich, "Which algorithms are feasible and which are not depends on the geometry of space-time", Geombinatorics, 1995, Vol. 4, No. 3, pp. 80-97.
file in pdf

PROJECTS: An important part of the class is a project. A project can be:

- reviewing and reporting on a related paper, or
- doing some independent research (not research as in high school, but research as in graduate school, i.e., trying to come up with something new), or
- programming something related to the class.

The most important aspect of the project is that it should be useful and/or interesting to *you*. The instructor can assign a project to you, there are plenty of potential projects, but if each student selects a project that he or she likes, this will be much much better for everyone.

TESTS: There will be three tests and the final exam.

GRADES: Each topic means home assignments (mainly on the sheets of paper, but some on the real computer). Some of them may be graded. Maximum number of points:

- first test: 10
- second test: 10
- third test: 15
- home assignments: 10
- final exam: 35
- project: 20

(smart projects with ideas that can turn into a serious scientific publication get up to 40 points).

A good project can help but it cannot completely cover possible deficiencies of knowledge as shown on the test and on the homeworks. In general, up to 80 points come from tests and home assignments. So:

- to get an A, you must gain, on all the tests and home assignments, at least 90% of the possible amount of points (i.e., at least 72), and also at least 90 points overall;
- to get a B, you must gain, on all the tests and home assignments, at least 80% of the possible amount of points (i.e., at least 64), and also at least 80 points overall;
- to get a C, you must gain, on all the tests and home assignments, at least 70% of the possible amount of points (i.e., at least 56), and also at least 70 points overall.

SPECIAL ACCOMMODATIONS: If you have a disability and need classroom accommodations, please contact the Center for Accommodations and Support Services (CASS) at 747-5148 or by email to cass@utep.edu, or visit their office located in UTEP Union East, Room 106. For additional information, please visit the CASS website at http://www.sa.utep.edu/cass. CASS's staff are the only individuals who can validate and if need be, authorize accommodations for students.

SCHOLASTIC DISHONESTY: Any student who commits an act of scholastic dishonesty is subject to discipline. Scholastic dishonesty includes, but not limited to cheating, plagiarism, collusion, submission for credit of any work or materials that are attributable to another person.

Cheating is:

- copying from the test paper of another student;
- communicating with another student during a test to be taken individually;
- giving or seeking aid from another student during a test to be taken individually;
- possession and/or use of unauthorized materials during tests (i.e. crib notes, class notes, books, etc.);
- substituting for another person to take a test;
- falsifying research data, reports, academic work offered for credit.

Plagiarism is:

- using someone's work in your assignments without the proper citations;
- submitting the same paper or assignment from a different course, without direct permission of instructors.

To avoid plagiarism see: https://www.utep.edu/student-affairs/osccr/_Files/docs/Avoiding-Plagiarism.pdf

Collusion is unauthorized collaboration with another person in preparing academic assignments.

Instructors are required to -- and will -- report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.

Notes: When in doubt on any of the above, please contact your instructor to check whether you are following an authorized procedure.

See you all in the Class!