68. Arbeitstagung für Allgemeine Algebra

Dresden, June 11, 2004

Complexity of Algebra and Algebra of Complexity

Mikhail Volkov

(Supported by the Alexander von Humboldt Foundation)

Ural State University, Ekaterinburg, Russia

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory:

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

These are classes of *combinatorial decision problems*,

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

These are classes of *combinatorial decision problems*, i. e. problems whose input is a finite object (graph, formula, algebra, . . .)

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

These are classes of *combinatorial decision problems*, i. e. problems whose input is a finite object (graph, formula, algebra, . . .)

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

These are classes of *combinatorial decision problems*, i. e. problems whose input is a finite object (graph, formula, algebra, . . .) and whose question is whether or not a given object possesses a certain property (which usually gives the name to the problem).

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

These are classes of *combinatorial decision problems*, i. e. problems whose input is a finite object (graph, formula, algebra, . . .) and whose question is whether or not a given object possesses a certain property (which usually gives the name to the problem).

5-minute Tour in Complexity Theory

The goal of the talk is to show a few striking examples of how fruitful the interaction between Algebra and Computational Complexity may be.

We start with recalling the fundamental notions of the latter theory: the classes **P** and **NP**.

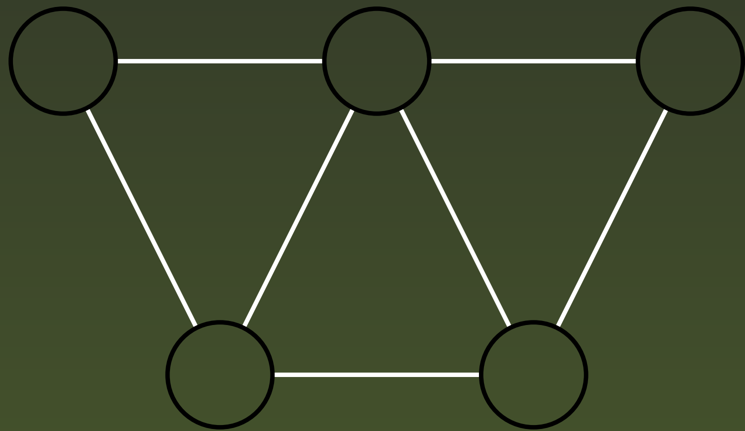
These are classes of *combinatorial decision problems*, i. e. problems whose input is a finite object (graph, formula, algebra, . . .) and whose question is whether or not a given object possesses a certain property (which usually gives the name to the problem). The answer to each concrete instance of such a problem is either YES or NO.

Example: Graph Coloring

The input of k-COLOR is a graph G.

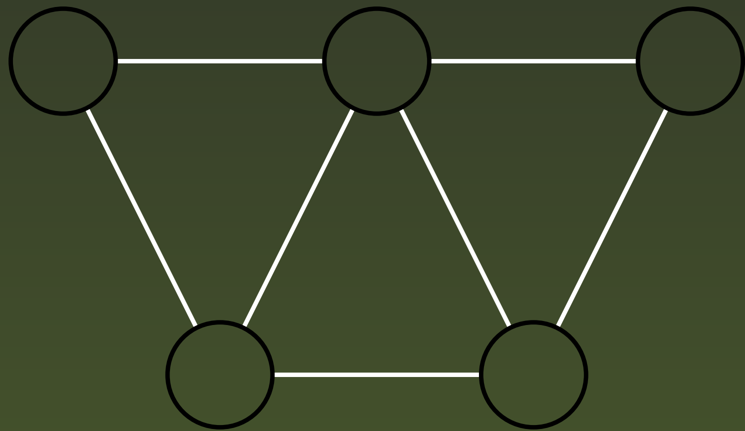
Example: Graph Coloring

The input of k-COLOR is a graph G.



Example: Graph Coloring

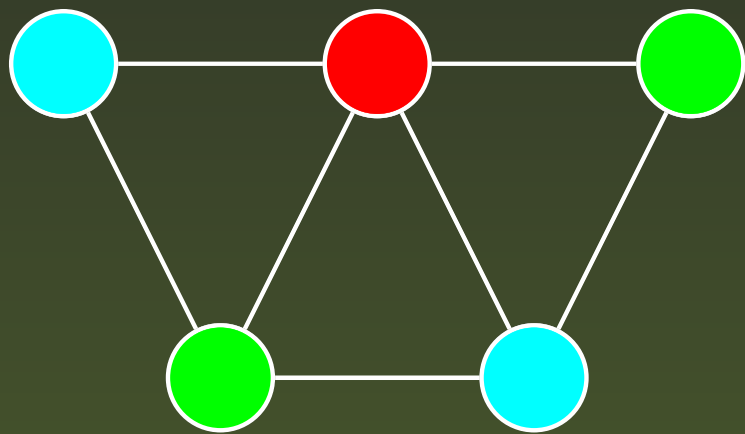
The input of k-COLOR is a graph G.



The question is whether the vertices of G can be labeled with k colors so that adjacent vertices are assigned different colors.

Example: Graph Coloring

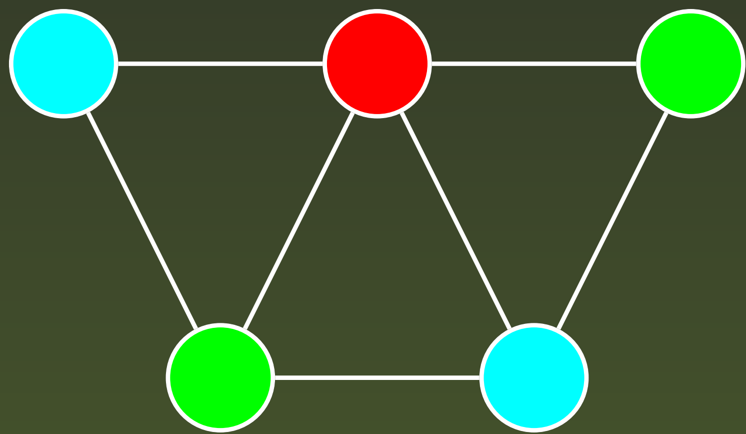
The input of k-COLOR is a graph G.



The question is whether the vertices of G can be labeled with k colors so that adjacent vertices are assigned different colors. For the above graph, the answer to 3-COLOR is YES

Example: Graph Coloring

The input of k-COLOR is a graph G.



The question is whether the vertices of G can be labeled with k colors so that adjacent vertices are assigned different colors. For the above graph, the answer to 3-COLOR is YES while the answer to 2-COLOR is NO.

The Classes P and NP

Arthur, a normal man

The Classes P and NP

Arthur, a normal man

Merlin, a superman

The Classes P and NP

A problem is in P if Arthur can solve it in polynomial time (of the size of its input).

The Classes P and NP

A problem is in **P** if Arthur can solve it in polynomial time (of the size of its input). Example: 2-COLOR is in **P** since Arthur can check in polynomial time whether or not all simple cycles of a given graph are of even length.

The Classes P and NP

A problem is in **P** if Arthur can solve it in polynomial time (of the size of its input). Example: 2-COLOR is in **P** since Arthur can check in polynomial time whether or not all simple cycles of a given graph are of even length.

A problem is in **NP** if, *whenever the answer to its instance is YES*, Merlin can convince Arthur that the answer is YES in polynomial time (of the size of the input).

The Classes P and NP

A problem is in **P** if Arthur can solve it in polynomial time (of the size of its input). Example: 2-COLOR is in **P** since Arthur can check in polynomial time whether or not all simple cycles of a given graph are of even length.

A problem is in **NP** if, *whenever the answer to its instance is YES*, Merlin can convince Arthur that the answer is YES in polynomial time (of the size of the input). Example: 3-COLOR is in **NP** since, given a 3-colorable graph, Merlin can exhibit its 3-coloring, and Arthur can check in polynomial time that this coloring is correct.

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$?

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth $1000000

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth \$1000000 (before tax).

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth \$1000000 (before tax). According to the present paradigm, we assume that $\mathbf{P} \neq \mathbf{NP}$.

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth \$1000000 (before tax). According to the present paradigm, we assume that $\mathbf{P} \neq \mathbf{NP}$.

An **NP**-*hard* problem is a problem to which any problem from **NP** can be reduced in polynomial time.

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth $1000000 (before tax). According to the present paradigm, we assume that $\mathbf{P} \neq \mathbf{NP}$.

An **NP**-*hard* problem is a problem to which any problem from **NP** can be reduced in polynomial time.

An **NP**-*complete* problem is a problem in **NP** that at the same time is **NP**-hard.

The Classes P and NP

Clearly $\mathbf{P} \subseteq \mathbf{NP}$. Is the inclusion strict? In other words, is $\mathbf{P} \neq \mathbf{NP}$? This is a VERY BIG PROBLEM which is worth \$1000000 (before tax). According to the present paradigm, we assume that $\mathbf{P} \neq \mathbf{NP}$.

An **NP**-*hard* problem is a problem to which any problem from **NP** can be reduced in polynomial time.

An **NP**-*complete* problem is a problem in **NP** that at the same time is **NP**-hard.

Example: 3-COLOR is **NP**-complete (Levin, 1973).

Complexity of Algebra

Many basic questions in algebra give rise to fascinating and sometimes very hard problems if one looks at these questions from the computational complexity standpoint.

Complexity of Algebra

Many basic questions in algebra give rise to fascinating and sometimes very hard problems if one looks at these questions from the computational complexity standpoint. For instance, given two finite algebras $\mathbf{A}$ and $\mathbf{B}$ of the same finite type, we know from Birkhoff's theorem that

$$\mathbf{B} \in \operatorname{var} \mathbf{A} \iff \mathbf{B} \text{ is a morphic image of } \mathrm{Clo}_{|B|}(\mathbf{A})$$

and the latter condition can be algorithmically tested.

Complexity of Algebra

Many basic questions in algebra give rise to fascinating and sometimes very hard problems if one looks at these questions from the computational complexity standpoint. For instance, given two finite algebras $\mathbf{A}$ and $\mathbf{B}$ of the same finite type, we know from Birkhoff's theorem that

$$\mathbf{B} \in \operatorname{var} \mathbf{A} \iff \mathbf{B} \text{ is a morphic image of } \mathrm{Clo}_{|B|}(\mathbf{A})$$

and the latter condition can be algorithmically tested.

But is this an efficient solution?

Complexity of Algebra

Many basic questions in algebra give rise to fascinating and sometimes very hard problems if one looks at these questions from the computational complexity standpoint. For instance, given two finite algebras $\mathbf{A}$ and $\mathbf{B}$ of the same finite type, we know from Birkhoff's theorem that

$$\mathbf{B} \in \text{var}\,\mathbf{A} \iff \mathbf{B} \text{ is a morphic image of } \mathrm{Clo}_{|B|}(\mathbf{A})$$

and the latter condition can be algorithmically tested.

But is this an efficient solution? It doesn't seem so — if $|A| = n$ and $|B| = m$, the only bound for the size of $\mathrm{Clo}_{|B|}(\mathbf{A})$ is $n^{(n^m)}$ so the above algorithm requires doubly exponential time (as a function of $|B|$).

Complexity of Algebra

Some important and application-oriented questions of formal language theory lead exactly to the problems of the form “Does a given $\mathbf{B}$ belong to the variety $\mathrm{var}\,\mathbf{A}$?" in which $\mathbf{A}$ and $\mathbf{B}$ are *finite semigroups*.

Complexity of Algebra

Some important and application-oriented questions of formal language theory lead exactly to the problems of the form “Does a given $\mathbf{B}$ belong to the variety $\mathrm{var}\,\mathbf{A}$?" in which $\mathbf{A}$ and $\mathbf{B}$ are *finite semigroups*.

Problem 1. *Does there exist a finite semigroup* $\mathbf{A}$ *such that testing membership in the variety* $\mathrm{var}\,\mathbf{A}$ *is* $\mathbf{NP}$-*complete?* $\mathbf{NP}$-*hard?* … $\mathbf{PSPACE}$-*complete?* $\mathbf{2\text{-}EXPTIME}$-*complete?*

Complexity of Algebra

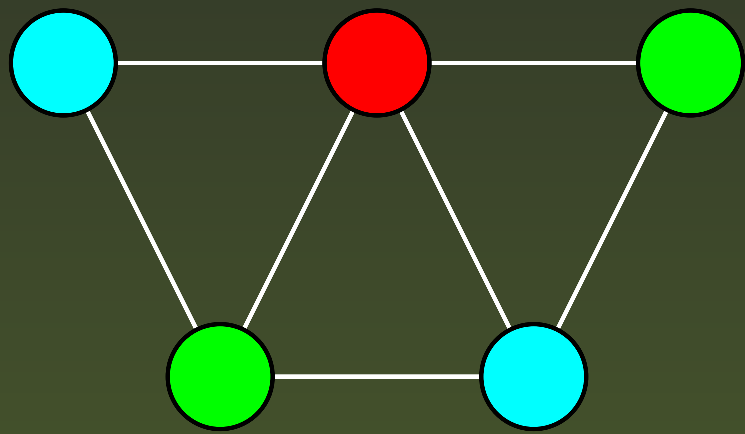
Some important and application-oriented questions of formal language theory lead exactly to the problems of the form “Does a given $\mathbf{B}$ belong to the variety $\mathrm{var}\,\mathbf{A}$?" in which $\mathbf{A}$ and $\mathbf{B}$ are *finite semigroups*.

Problem 1. *Does there exist a finite semigroup* $\mathbf{A}$ *such that testing membership in the variety* $\mathrm{var}\,\mathbf{A}$ *is* $\mathbf{NP}$*-complete?* $\mathbf{NP}$*-hard?* $\ldots$ $\mathbf{PSPACE}$*-complete?* $\mathbf{2}$-$\mathbf{EXPTIME}$*-complete?*

Ralph McKenzie found a partial solution of this problem which was then refined by Marcel Jackson. The solution will appear in their joint paper in “International Journal of Algebra and Computation”.

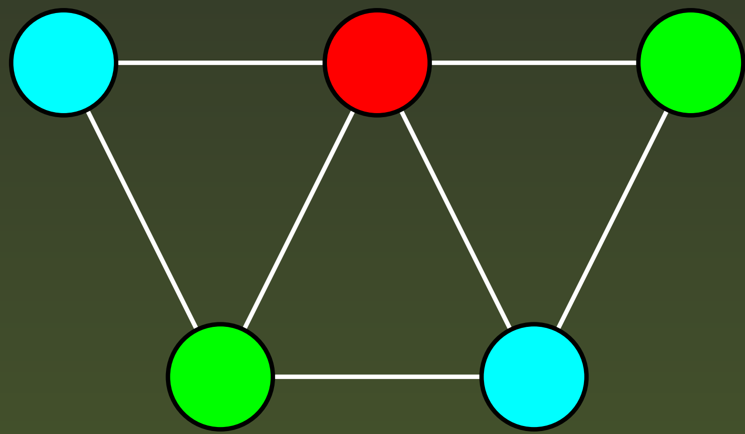
McKenzie–Jackson Theorem

Let C_3 be the graph used as an example 22 slides ago:



McKenzie–Jackson Theorem

Let C_3 be the graph used as an example 22 slides ago:



McKenzie and Jackson assign to each finite graph G a finite semigroup $S(G)$ such that $S(G) \in \operatorname{var} S(C_3)$ iff G belongs to the universal Horn class generated by C_3.

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs.

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs. Hence $S(G) \in \operatorname{var} S(C_3)$ iff the graph G is 3-colorable.

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs. Hence $S(G) \in \operatorname{var} S(C_3)$ iff the graph G is 3-colorable. Since 3-COLOR is **NP**-complete, we conclude that the membership problem for the variety $\operatorname{var} S(C_3)$ is **NP**-hard.

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs. Hence $S(G) \in \operatorname{var} S(C_3)$ iff the graph G is 3-colorable. Since 3-COLOR is $\mathbf{NP}$-complete, we conclude that the membership problem for the variety $\operatorname{var} S(C_3)$ is **$\mathbf{NP}$-hard**. (One can't yet claim that the problem is $\mathbf{NP}$-complete because it is not yet clear that it belongs to the class $\mathbf{NP}$.)

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs. Hence $S(G) \in \mathrm{var}\, S(C_3)$ iff the graph G is 3-colorable. Since 3-COLOR is **NP**-complete, we conclude that the membership problem for the variety $\mathrm{var}\, S(C_3)$ is **NP**-hard. (One can't yet claim that the problem is **NP**-complete because it is not yet clear that it belongs to the class **NP**.)

If the graph G has n vertices, than the semigroup $S(G)$ has $n^2 + 5n + 5$ elements. In particular, the semigroup $S(C_3)$ has 55 elements.

McKenzie–Jackson Theorem

However, it is known (Nešetřil and Pultr, 1978) that the universal Horn class generated by C_3 is precisely the class of all 3-colorable graphs. Hence $S(G) \in \operatorname{var} S(C_3)$ iff the graph G is 3-colorable. Since 3-COLOR is **NP**-complete, we conclude that the membership problem for the variety $\operatorname{var} S(C_3)$ is **NP**-hard. (One can't yet claim that the problem is **NP**-complete because it is not yet clear that it belongs to the class **NP**.)

If the graph G has n vertices, than the semigroup $S(G)$ has $n^2 + 5n + 5$ elements. In particular, the semigroup $S(C_3)$ has 55 elements. The structure of the semigroup $S(G)$ is rather transparent: it is a *local semilattice*.

Application to the Finite Basis Problem

A semigroup **A** is said to be *finitely based* if all identities holding in **A** follow from a finite set of such identities (an *identity basis* of **A**).

Application to the Finite Basis Problem

A semigroup $\mathbf{A}$ is said to be *finitely based* if all identities holding in $\mathbf{A}$ follow from a finite set of such identities (an *identity basis* of $\mathbf{A}$). If we know a finite identity basis Φ of a semigroup $\mathbf{A}$ then we can use it to efficiently decide the membership in $\operatorname{var} \mathbf{A}$.

Application to the Finite Basis Problem

A semigroup $\mathbf{A}$ is said to be *finitely based* if all identities holding in $\mathbf{A}$ follow from a finite set of such identities (an *identity basis* of $\mathbf{A}$). If we know a finite identity basis Φ of a semigroup $\mathbf{A}$ then we can use it to efficiently decide the membership in $\mathrm{var}\,\mathbf{A}$. Indeed, given a finite semigroup B, we can simply check if it satisfies each identity in Φ, and this requires polynomial time (as a function of $|B|$).

Application to the Finite Basis Problem

A semigroup $\mathbf{A}$ is said to be *finitely based* if all identities holding in $\mathbf{A}$ follow from a finite set of such identities (an *identity basis* of $\mathbf{A}$). If we know a finite identity basis Φ of a semigroup $\mathbf{A}$ then we can use it to efficiently decide the membership in $\operatorname{var}\mathbf{A}$. Indeed, given a finite semigroup $\mathbf{B}$, we can simply check if it satisfies each identity in Φ, and this requires polynomial time (as a function of $|B|$).

Comparing this with the McKenzie–Jackson Theorem, we conclude that the 55-element semigroup $S(C_3)$ is non-finitely based.

Application to the Finite Basis Problem

A semigroup $\mathbf{A}$ is said to be *finitely based* if all identities holding in $\mathbf{A}$ follow from a finite set of such identities (an *identity basis* of $\mathbf{A}$). If we know a finite identity basis Φ of a semigroup $\mathbf{A}$ then we can use it to efficiently decide the membership in $\operatorname{var}\mathbf{A}$. Indeed, given a finite semigroup B, we can simply check if it satisfies each identity in Φ, and this requires polynomial time (as a function of $|B|$).

Comparing this with the McKenzie–Jackson Theorem, we conclude that the 55-element semigroup $S(C_3)$ is non-finitely based. This does not follow from any known result on the finite basis problem for finite semigroups!

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!

Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short).

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!
Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short). The aim in a CSP is to assign values to a given set of variables such that certain constraints are satisfied.

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!
Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short). The aim in a CSP is to assign values to a given set of variables such that certain constraints are satisfied.

Example: k-COLOR. Here

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!
Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short). The aim in a CSP is to assign values to a given set of variables such that certain constraints are satisfied.

Example: k-COLOR. Here

- variables are the vertices of a given graph;

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!
Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short). The aim in a CSP is to assign values to a given set of variables such that certain constraints are satisfied.

Example: k-COLOR. Here

- variables are the vertices of a given graph;
- values are the colors;

Algebra of Complexity

The bridges between Algebra and Computational Complexity are open in both directions!
Many combinatorial decision problems can be naturally thought of as Constraint Satisfaction Problems (CSP, for short). The aim in a CSP is to assign values to a given set of variables such that certain constraints are satisfied.

Example: k-COLOR. Here

- variables are the vertices of a given graph;
- values are the colors;
- constraints are determined by the edges of the graph: variables corresponding to adjacent vertices should be assigned different values.

Constraint Satisfaction Problems

Consider the family of CSPs in which values are taken from a finite set D (*domain*) and the constraint relations are taken from a set Γ of relations on D.

Constraint Satisfaction Problems

Consider the family of CSPs in which values are taken from a finite set D (*domain*) and the constraint relations are taken from a set Γ of relations on D. This collection is denoted by $\mathrm{CSP}(\Gamma)$.

Constraint Satisfaction Problems

Consider the family of CSPs in which values are taken from a finite set D (*domain*) and the constraint relations are taken from a set Γ of relations on D. This collection is denoted by $\mathrm{CSP}(\Gamma)$.

For instance, 3-COLOR can be easily identified with $\mathrm{CSP}(\Gamma)$ where the domain D is $\{$red, green, blue$\}$ and Γ consists of the inequality relation $\neq_D$ on D.

Constraint Satisfaction Problems

Consider the family of CSPs in which values are taken from a finite set D (*domain*) and the constraint relations are taken from a set Γ of relations on D. This collection is denoted by $\mathrm{CSP}(\Gamma)$.

For instance, 3-COLOR can be easily identified with $\mathrm{CSP}(\Gamma)$ where the domain D is {red, green, blue} and Γ consists of the inequality relation $\neq_D$ on D. In particular, this shows that generally speaking $\mathrm{CSP}(\Gamma)$ is **NP**-complete.

Constraint Satisfaction Problems

An important direction of research in Computational Complexity groups around the following

Constraint Satisfaction Problems

An important direction of research in Computational Complexity groups around the following

Problem 2. *For which relation sets* Γ *the family* $\mathrm{CSP}(\Gamma)$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*–complete?*

Constraint Satisfaction Problems

An important direction of research in Computational Complexity groups around the following

Problem 2. *For which relation sets* Γ *the family* $\mathrm{CSP}(\Gamma)$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*–complete?*

A very successful purely algebraic approach to this problem has been recently developed by Peter Jeavons, Andrei Bulatov, Andrei Krokhin e.a.

Constraint Satisfaction Problems

An important direction of research in Computational Complexity groups around the following
Problem 2. *For which relation sets* Γ *the family* $\mathrm{CSP}(\Gamma)$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*–complete?*

A very successful purely algebraic approach to this problem has been recently developed by Peter Jeavons, Andrei Bulatov, Andrei Krokhin e.a. First it has been observed (Jeavons, 1998) that for any relation set Γ the families $\mathrm{CSP}(\Gamma)$ and $\mathrm{CSP}(\mathrm{Clo}(\Gamma))$ have the same complexity.

Constraint Satisfaction Problems

An important direction of research in Computational Complexity groups around the following
Problem 2. *For which relation sets* Γ *the family* $\mathrm{CSP}(\Gamma)$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*–complete?*

A very successful purely algebraic approach to this problem has been recently developed by Peter Jeavons, Andrei Bulatov, Andrei Krokhin e.a. First it has been observed (Jeavons, 1998) that for any relation set Γ the families $\mathrm{CSP}(\Gamma)$ and $\mathrm{CSP}(\mathrm{Clo}(\Gamma))$ have the same complexity. Here $\mathrm{Clo}(\Gamma)$ is the *relational clone* generated by Γ.

Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D.

Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D. Recall that there is a well understood and very useful Galois connection between the relational clones on the set D and the clones of operations over D, cf. Pöschel-Kalužnin, 1979, and/or A. Szendrei, 1986.

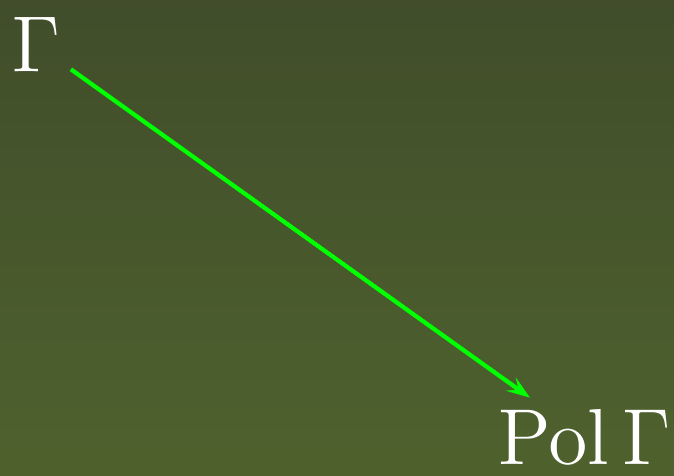
Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D. Recall that there is a well understood and very useful Galois connection between the relational clones on the set D and the clones of operations over D, cf. Pöschel-Kalužnin, 1979, and/or A. Szendrei, 1986.

$$\Gamma$$

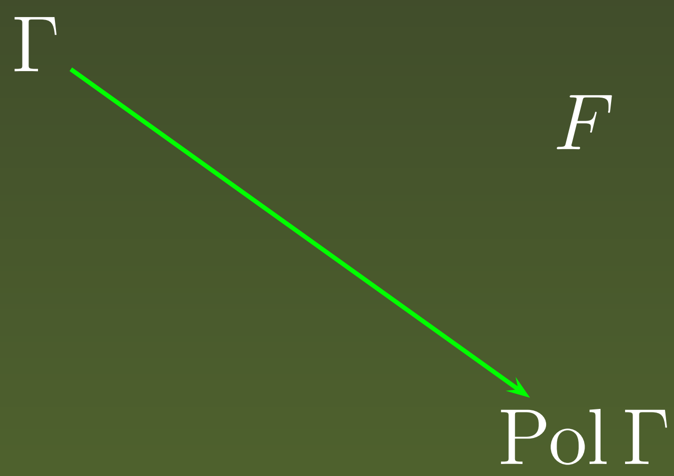
Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D. Recall that there is a well understood and very useful Galois connection between the relational clones on the set D and the clones of operations over D, cf. Pöschel-Kalužnin, 1979, and/or A. Szendrei, 1986.



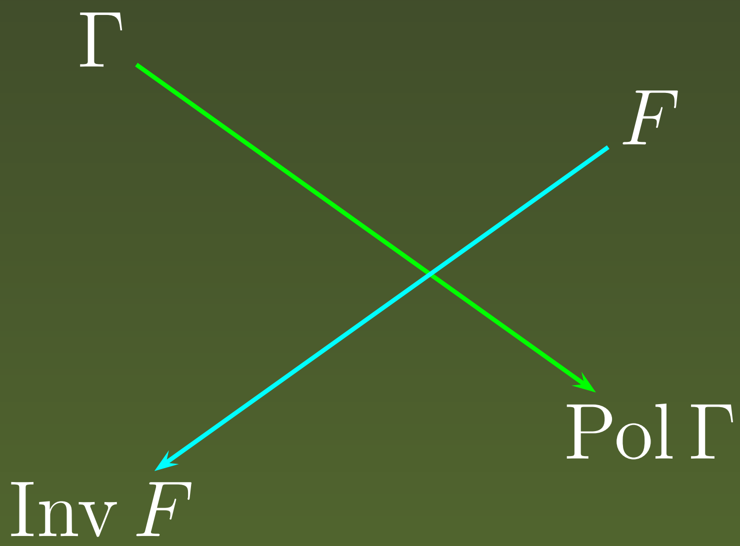
Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D. Recall that there is a well understood and very useful Galois connection between the relational clones on the set D and the clones of operations over D, cf. Pöschel-Kalužnin, 1979, and/or A. Szendrei, 1986.



Relations and Operations

Thus, it suffices to study families $\mathrm{CSP}(\Gamma)$ such that Γ is a clone of relations on the domain D. Recall that there is a well understood and very useful Galois connection between the relational clones on the set D and the clones of operations over D, cf. Pöschel-Kalužnin, 1979, and/or A. Szendrei, 1986.



Relations and Operations

Thus, each finite algebra $\mathbf{D} = \langle D, F\rangle$ on the domain D defines a family of CSPs, namely, $\mathrm{CSP}(\mathrm{Inv}\, F)$.

Relations and Operations

Thus, each finite algebra $\mathbf{D} = \langle D, F \rangle$ on the domain D defines a family of CSPs, namely, $\mathrm{CSP}(\mathrm{Inv}\, F)$. We denote this family by $\mathrm{CSP}(\mathrm{Inv}\, \mathbf{D})$.

Relations and Operations

Thus, each finite algebra $\mathbf{D} = \langle D, F \rangle$ on the domain D defines a family of CSPs, namely, $\mathrm{CSP}(\mathrm{Inv}\, F)$. We denote this family by $\mathrm{CSP}(\mathrm{Inv}\, \mathbf{D})$.

Thus, the algebraic counterpart of Problem 2 is:

Problem 3. *For which finite algebras* $\mathbf{D}$ *the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*-complete?*

Relations and Operations

Thus, each finite algebra $\mathbf{D} = \langle D, F \rangle$ on the domain D defines a family of CSPs, namely, $\mathrm{CSP}(\mathrm{Inv}\, F)$. We denote this family by $\mathrm{CSP}(\mathrm{Inv}\, \mathbf{D})$.
Thus, the algebraic counterpart of Problem 2 is:

Problem 3. *For which finite algebras* $\mathbf{D}$ *the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$*? is* $\mathbf{NP}$*-complete?*

It turns out that the complexity of $\mathrm{CSP}(\mathbf{D})$ depends on some deep algebraic properties of $\mathbf{D}$, in particular, on the Tame Congruence Theory labeling of its congruence lattice.

CSPs Defined by Semigroups

I conclude with a sample result (Bulatov, Jeavons, $\sim$, 2002):

CSPs Defined by Semigroups

I conclude with a sample result (Bulatov, Jeavons, $\sim$, 2002): *for a finite semigroup* $\mathbf{D}$, *the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$ *iff* $\mathbf{D}$ *satisfies the quasi-identities*

$$ef = e^2 = e \;\&\; fe = f^2 = f \Rightarrow e = f,$$

$$ef = f^2 = f \;\&\; fe = e^2 = e \Rightarrow e = f.$$

CSPs Defined by Semigroups

I conclude with a sample result (Bulatov, Jeavons, $\sim$, 2002): *for a finite semigroup* $\mathbf{D}$, *the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$ *iff* $\mathbf{D}$ *satisfies the quasi-identities*

$$ef = e^2 = e \mathbin{\&} fe = f^2 = f \Rightarrow e = f,$$

$$ef = f^2 = f \mathbin{\&} fe = e^2 = e \Rightarrow e = f.$$

Otherwise the family $\mathrm{CSP}(\mathbf{D})$ *is* $\mathbf{NP}$*-complete.*

CSPs Defined by Semigroups

I conclude with a sample result (Bulatov, Jeavons, $\sim$, 2002): *for a finite semigroup* $\mathbf{D}$, *the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$ *iff* $\mathbf{D}$ *satisfies the quasi-identities*

$$ef = e^2 = e \;\&\; fe = f^2 = f \Rightarrow e = f,$$
$$ef = f^2 = f \;\&\; fe = e^2 = e \Rightarrow e = f.$$

Otherwise the family $\mathrm{CSP}(\mathbf{D})$ *is* $\mathbf{NP}$*-complete.*

Finite semigroups satisfying the above quasi-identities are known in semigroup theory as *block-groups*.

CSPs Defined by Semigroups

I conclude with a sample result (Bulatov, Jeavons, $\sim$, 2002): *for a finite semigroup* $\mathbf{D}$*, the family* $\mathrm{CSP}(\mathbf{D})$ *is in* $\mathbf{P}$ *iff* $\mathbf{D}$ *satisfies the quasi-identities*

$$ef = e^2 = e \,\&\, fe = f^2 = f \Rightarrow e = f,$$
$$ef = f^2 = f \,\&\, fe = e^2 = e \Rightarrow e = f.$$

Otherwise the family $\mathrm{CSP}(\mathbf{D})$ *is* $\mathbf{NP}$*-complete.*

Finite semigroups satisfying the above quasi-identities are known in semigroup theory as *block-groups*. They are known to play a distinguished role in algebraic theory of formal languages.

Conclusion and Announcement

What we have seen is just a few samples from a quickly developing and promising area.

Conclusion and Announcement

What we have seen is just a few samples from a quickly developing and promising area. Today there will be two further talks at our conference devoted to its other aspects:

Conclusion and Announcement

What we have seen is just a few samples from a quickly developing and promising area. Today there will be two further talks at our conference devoted to its other aspects:

- the talk by Ondřej Klima on Complexity issues of checking identities in finite monoids (A317, $10^{45} - 11^{00}$) — it belongs to the Complexity of Algebra direction;

Conclusion and Announcement

What we have seen is just a few samples from a quickly developing and promising area. Today there will be two further talks at our conference devoted to its other aspects:

- the talk by Ondřej Klima on Complexity issues of checking identities in finite monoids (A317, $10^{45} - 11^{00}$) — it belongs to the Complexity of Algebra direction;
- the talk by László Zadori on Bounded width algebras (B321, $15^{05} - 15^{20}$) — it belongs to the Algebra of Complexity direction.

One More Announcement

I am honored to announce that the Ural State University and the Institute of Mathematics and Mechanics of the Ural Branch of the Russian Academy of Sciences organize an International Algebraic Conference in Ekaterinburg from August 29 to September 3, 2005.

One More Announcement

I am honored to announce that the Ural State University and the Institute of Mathematics and Mechanics of the Ural Branch of the Russian Academy of Sciences organize an International Algebraic Conference in Ekaterinburg from August 29 to September 3, 2005. The conference is dedicated to the centennial of the birthday of Professor Piotr Grigor'evich Kontorovich, the founder of the Sverdlovsk (=Ekaterinburg) algebraic school,

One More Announcement

I am honored to announce that the Ural State University and the Institute of Mathematics and Mechanics of the Ural Branch of the Russian Academy of Sciences organize an International Algebraic Conference in Ekaterinburg from August 29 to September 3, 2005. The conference is dedicated to the centennial of the birthday of Professor Piotr Grigor'evich Kontorovich, the founder of the Sverdlovsk (=Ekaterinburg) algebraic school, and to the 70th birthday of Lev Naumovich Shevrin.

One More Announcement

The topics of the conference will include all major areas of general algebra (theories of semigroups, groups, rings, lattices, universal algebra) and model theory.

One More Announcement

The topics of the conference will include all major areas of general algebra (theories of semigroups, groups, rings, lattices, universal algebra) and model theory. During the conference, on September 1, 2005, L.N.Shevrin's seminar "Algebraic Systems" will celebrate its 1000th meeting.

One More Announcement

The topics of the conference will include all major areas of general algebra (theories of semigroups, groups, rings, lattices, universal algebra) and model theory. During the conference, on September 1, 2005, L.N.Shevrin's seminar "Algebraic Systems" will celebrate its 1000th meeting.
The first announcement will be distributed at the beginning of July.

One More Announcement

The topics of the conference will include all major areas of general algebra (theories of semigroups, groups, rings, lattices, universal algebra) and model theory. During the conference, on September 1, 2005, L.N.Shevrin's seminar "Algebraic Systems" will celebrate its 1000th meeting.
The first announcement will be distributed at the beginning of July.

I am looking forward to seeing all of you, dear friends, in Ekaterinburg next year!

One More Announcement