Lecture 12

What we did in Lecture 11

- We studied the distinctive power of graph polynomials and graph parameters
- We studied examples
- We introduced a comparison of graph polynomials via their coefficients
- We showed the equivalence of the two notions of comparability
- We studied more examples

Out line of Lecture 12

- Some computations of graph polynomials
- Discovering linear recurrence relations
- Recursively defined families of graphs
- $\mathbf{MSOL}$-definable graph polynomials
- A general theorem about linear recurrence relations for recursively defined families of graphs and $\mathbf{MSOL}$-definable graph polynomials

Main theme

We deal with a purely **graph theoretical** problem:

Given a **regularly constructed indexed family of graphs** G_n such as
the paths P_n, the circles C_n, the wheels W_n, the cliques K_n, the grids $Grid_{m,n}$

and a **graph polynomial** P, such as
the matching , Tutte, clique, cover polynomial

compute all the values $P(G_n)$.

Often we have a (linear) recurrence relation, i.e. there is $q \in \mathbb{N}$,
and polynomials $p_1, \ldots, p_q \in \mathbb{Z}[\bar{X}]$ such that for sufficiently large n

$$P(G_{n+q+1}) = \sum_{i=1}^{q} p_i \cdot P(G_{n+i})$$

When is this the case?

We shall see that methods from **LOGIC** help clarifying the situation.

Case study: The chromatic polynomial

For a graph G, the **chromatic polynomial** $\chi(G, \lambda) \in \mathbb{Z}[\lambda]$
is the polynomial such that for $t \in \mathbb{N}$
the value of $\chi(G, t)$ is the number of t-vertex colorings of G.

Let P_n be the path and K_n the complete graph on n vertices,

We compute $\chi(P_n, \lambda)$ and $\chi(K_n, \lambda)$.

- $\chi(P_1, \lambda) = \lambda$ and $\chi(P_2, \lambda) = \lambda \cdot (\lambda - 1)$.
- $\chi(P_{n+1}, \lambda) = \chi(P_n, \lambda) \cdot (\lambda - 1)$.
- $\chi(K_{n+1}, \lambda) = \lambda \cdot (\lambda - 1) \ldots (\lambda - n) = (\lambda - n)\chi(K_n - 1)$

For P_n we have a **linear recurrence relation independently of** n,
for K_n the recurrence **depends on** n. Can we find one **not** depending on n?

Homework

Compute the chromatic polynomial for the following families of graphs:

- The circles C_n;
- The wheels W_n;
- The ladders L_n;
- The grids $Grid_{n,m}$.

Case study: The clique and independent set polynomial

For a graph G, the **clique polynomial** $cl(G, X) \in \mathbb{Z}[X]$ is defined by

$$cl(G, X) = \sum cl_k(G) \cdot X^k$$

where $cl_k(G)$ is the number of k-cliques of G.

We compute $cl(P_n, \lambda)$ and $cl(K_n, \lambda)$:

$cl(P_{n+1})(X) = (1 + X) + cl(P_n)(X)$,
$cl(K_{n+1})(X) = \sum_k^{n+1} \binom{n}{k} X^k = (X + 1)^{n+1} = (X + 1) \cdot cl(K_n)(X)$

The **independent set polynomial** $in(G, X) \in \mathbb{Z}[X]$ is defined by

$$in(G, X) = \sum in_k(G) \cdot X^k$$

where $in_k(G)$ is the number of independent sets of G of size k.

Homework: Compute the linear recurrences for $in(P_n)(X)$ and $in(K_n)(X)$.

Case Study: The matching polynomial, I

For a graph G, the **matching polynomial** $\mu(G, X) \in \mathbb{Z}[X]$ is defined by

$$\mu(G, X) = \sum m_k(G) \cdot X^k$$

where $m_k(G)$ is the number of k-matchings of G.

We compute $\mu(P_n, X)$:

We use auxiliary polynomials

$$\mu^+(P_n, X) = \sum m_k^+(P_n) \cdot X^k$$

and

$$\mu^-(P_n, X) = \sum m_k^-(P_n) \cdot X^k$$

where $m_k^+(P_n)$ and $m_k^-(P_n)$ is the number of k-matchings of P_n which **includes**, respectively **excludes** the last vertex.

Clearly we have $m_k(P_n) = m_k^+(P_n) + m_k^-(P_n)$ hence

$$\mu(P_n, X) = \mu^+(P_n, X) + \mu^-(P_n, X)$$

Case study: The matching polynomial, II

It is easy to see that

$$\mu^{-}(P_{n+1}) = \mu^{-}(P_n) + \mu^{+}(P_n)$$
$$\mu^{+}(P_{n+1}) = X \cdot \mu^{-}(P_n)$$

For $\bar{\mu}_n = (\mu^{-}(P_n), \mu^{+}(P_n))^t$ we get

$$A\bar{\mu}_n = \bar{\mu}_{n+1}$$

with

$$a_{1,1} = 1, a_{1,2} = 1, a_{2,1} = X, a_{2,2} = 0$$

The characteristic polynomial of A is

$$det(\lambda \mathbf{1} - A) = \lambda^2 - \lambda - X$$

so we get the linear recurrence relation (independent of n)

$$\mu(P_{n+2}) = \mu(P_{n+1}) + X \cdot \mu(P_n)$$

Case study: The vertex-cover polynomial

For a graph G, the **vertex-cover polynomial** $vc(G, X) \in \mathbb{Z}[X]$ is defined by

$$vc(G, X) = \sum vc_k(G) \cdot X^k$$

where $vc_k(G)$ is the number of k-vertex-covers of G.

- $vc(P_{n+1}, X) = X \cdot vc(P_n, X) + X \cdot vc(P_{n-1}, X)$
- $vc(C_{n+1}, X) = X \cdot vc(C_n, X) + X^2 \cdot vc(C_{n-2}, X)$
- Let L_n be the graph which consists of n isolated loops.
  $vc(L_{n+1}, X) = X \cdot vc(L_n, X) = X^n$
- For the wheel graph W_n we have
  $vc(W_{n+1}, X) = X \cdot vc(W_n, X) + X^n = X \cdot vc(W_n, X) + X \cdot vc(L_n, X)$
  hence, using the characteristic polynomial of the matrix, $A = (a_{i,j})$ with
  $a_{1,1} = a_{1,2} = a_{2,2} = X$ and $a_{2,1} = 0$
  $vc(W_{n+1}, X) = 2X \cdot vc(W_n, X) - X^2 \cdot vc(W_{n-1}, X)$

F.M. Dong, M.D. Hendy, K.L. Teo and C.H.C. Little, The vertex-cover polynomial of a graph, Discrete Mathematics 250 (2002), 71-78

P-recursive families of graphs, I

Definition 1 *Let P be a graph polynomial and $\mathcal{G} = \{G_n : n \in \mathbb{N}\}$ be a family of graphs. $\mathcal{G}$ is said to be* **P-recursive** *if there is $q \in \mathbb{N}$, and polynomials $p_1, \ldots, p_q \in \mathbb{Z}[\bar{X}]$ such that for sufficiently large n*

$$P(G_{n+q+1}) = \sum_{i=1}^{q} p_i \cdot P(G_{n+i})$$

Let P_n be the path on n vertices. We get, for sufficiently large n,

- for the chromatic polynomials: $c(P_{n+1})(\lambda) = c(P_n, \lambda)(\lambda) \cdot (\lambda - 1)$.
- for the clique polynomials:
  $cl(P_{n+1})(X) = (1 + X) + cl(P_n)(X)$,
  $cl(K_{n+1})(X) = \sum_k^{n+1} \binom{n}{k} X^k = (X + 1)^{n+1} = (X + 1) \cdot cl(K_n)(X)$
- for the matching polynomials: $\mu(P_{n+1})(X) = X \cdot \mu(P_{n-1})(X) + \mu(P_n)(X)$,
- for the Tutte polynomials: $T(P_{n+1})(X, Y) = Y \cdot T(P_n)(X, Y)$.
- for the vertex-cover polynomials: $vc(P_{n+1}, X) = X \cdot vc(P_n, X) + X \cdot vc(P_{n-1}, X)$

P-recursive families of graphs, II

The characteristic series $\sigma_P(\mathcal{G})$ of $\mathcal{G}$ is defined as

$$\sigma_P(\mathcal{G}) = \sum_{i=1} P(G_i) Z^i.$$

Proposition 2 (Folklore) *$\mathcal{G}$ is P-**recursive** iff the characteristic series*

$$\sigma_P(\mathcal{G})$$

is a rational function in $\bar{X}$ and Z.

T-recursive families, I

N.L. Biggs, R.M. Damerell and D.A. Sands, 1972

In 1972 N.L. Biggs, R.M. Damerell and D.A. Sands introduced recursive families of graphs.

These are our T-recursive families of graphs where T is the Tutte polynomial.

They show that several families of graphs are recursive (in their sense). Among them there are:

cycles, ladders and wheels

All these families have in common that they can be constructed from an initial graph by the repeated application of a fixed graph operation

N.L. Biggs, R.M. Damerell and D.A. Sands,
Recursive families of graphs, J. Combin. Theory Ser. B 12 (1972), 123-131

T-recursive families, II

M. Noy and A. Ribó, 2004

In 2004 M. Noy and A. Ribó study which graph families G_n,
constructed from an initial graph G_0,
by the repeated application of a fixed graph operation $F(G)$,
are T-**recursive families of graphs**.

They introduce a notion of

recursively constructible families of graphs,

and show that every such family is T-recursive.

Their notion is reminiscent of certain graph grammars.

M. Noy and A. Ribó, Recursively constructible families of graphs, Advances in Applied Mathematics 32 (2004) 350-363.

Using Logic

We use the finite model theory
of Monadic Second Order Logic (MSOL)
to extend these results in several ways:

- We prove that for every P from a wide class of graph polynomials, the MSOL-definable graph polynomials, every recursively constructible family G_n is P-recursive.
- We extend the result to the class of iteration families of graphs which is proper extension of the class of recursively constructible families.
- We extend the result to signed graphs and knot diagrams and to various knot polynomials.
- We extend the result to hypergraphs and relational structures.

Iteration families, I

In the absence of the formalisms of graph grammars Noy and Ribó give an adhoc definition of

repeated fixed succession
of elementary operations

which can be applied to a graph with a *context*, i.e. a labeled graph.

Definition 3 *Let F denote such an operation.*

Given a graph (with context) G, we put

$$G_0 = G, G_{n+1} = F(G_n)$$

Then the family

$$\mathcal{G} = \{G_n : n \in \mathbb{N}\}$$

is called **recursively constructible** *using F, or an* F**-iteration family**.

Iteration families, II

Given a graph polynomial P,

the question now is to find

a characterization of those operations F,

for which a linear recurrence for the polynomials $P(G_n)$ holds.

M. Noy and A. Ribó give only a **suffient condition**
for the case of the **Tutte polynomial**.

General strategy

We proceed as outlined in the case of the matching and the vertex-cover polynomial.

To compute $P(G_{n+1})$, we try to find,

depending on P and, possibly, on G_0, but **independently of** n

- an $m \in \mathbb{N}$,
- auxiliary polynomials $P_i(G_{n+1}), i \leq m$,
- and a matrix $A = (a_{i,j}) \in \mathbb{Z}[\bar{X}]^{m \times m}$

such that

$$P_j(G_{n+1})(\bar{X}) = \sum_i a_{i,j}(\bar{X}) \cdot P_i(G_n)(\bar{X})$$

Then we use the **characteristic polynomial of the matrix** A

to convert this into a **linear recurrence** relation.

Where logic enters for the graph polynomial P?

Definition 4 *A polynomial P of the form*

$$P(G) = \sum_{(V,E') \in K_1} \left(\prod_{E' \subseteq E} t(\bar{X}) \right)$$

or

$$P(G) = \sum_{(V',E|V') \in K_2} \left(\prod_{V' \subseteq V} t(\bar{X}) \right)$$

where $t(\bar{X})$ is a fixed term in the indeterminates $\bar{X}$ and K_1 or K_2 are definable in **Monadic Second Order Logic (MSOL)** *is called an*

$MSOL$-**definable graph polynomials**.

There are **more general** versions of this definition, but **here** this suffices.

$MSOL$-definable polynomials, I

The **clique and independent set polynomials** are **of the form**

$$P(G) = \sum_{(V', E|V') \in K_2} \left(\prod_{V' \subseteq V} X \right)$$

because saying that the induced subgraph $G[V']$ is a clique or an independent set is $MSOL$-definable.

Rearranging the terms we get

$$cl(G) = \sum_{(V', E|V') \in Clique} \left(\prod_{V' \subseteq V} X \right) = \sum_{k} cl_k(G) \cdot X^k$$

and

$$in(G) = \sum_{(V', E|V') \in Indep} \left(\prod_{V' \subseteq V} X \right) = \sum_{k} in_k(G) \cdot X^k$$

Note: The second order variable for V' is needed.

$MSOL$-definable polynomials, II

The **vertex-cover polynomials** are **of the form**

$$P(G) = \sum_{(V,E,V') \in K_2} \left(\prod_{V' \subseteq V} X \right)$$

because saying that V' is a vertex-cover of (V, E) is $MSOL$-definable.

Rearranging the terms we get

$$vc(G) = \sum_{(V,E,V') \in VC} \left(\prod_{V' \subseteq V} X \right) = \sum_{k} vc_k(G) \cdot X^k$$

Note: The second order variable for V' is again needed.

$MSOL$-definable polynomials, III

The **generating matching polynomials** are **of the form**

$$gm(G) = \sum_{(V,E') \in Matching} \left(\prod_{E' \subseteq E} X \right)$$

However, being a matching is

- **NOT** $MSOL$-definable if graphs are represented as $G = (V, E)$.
- but **IS** $MSOL$-definable, if the graph is represented by its incidence graph $I(G) = (V \cup E, R)$.

For the **Tutte polynomial**, we have to add a linear order on the edges, to make it $MSOL$-definable, and note, that the Tutte polynomial is then indepent of the order on the edges.

Where logic enters for the operation F?

Ehrenfeucht games again

Let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-structures.

Recall: We write $\mathfrak{A} \equiv_q^{MSOL} \mathfrak{B}$, if $\mathfrak{A}$ and $\mathfrak{B}$ cannot be distinguished by $MSOL(\tau)$-formulas of quantifier rank q.

Definition 5
An operation F on τ-structures is **$MSOL$-smooth**
if whenever $\mathfrak{A} \equiv_q^{MSOL} \mathfrak{B}$, then also $F(\mathfrak{A}) \equiv_q^{MSOL} F(\mathfrak{B})$.

The operation F should be $MSOL$-smooth for the **presentation of the graphs**, for which the polynomial is $MSOL$-definable.

$\mathbf{MSOL}$-smooth operations: Examples

Here graphs are of the form $G = (V(G), E(G))$.

Let $H = (V(H), E(H))$ be a fixed graph.

- The operation $D_H(G) = G \sqcup H$ is $\mathbf{MSOL}$-smooth.
- The operation $J_H(G) = G \bowtie H$ is $\mathbf{MSOL}$-smooth.
- The cliques K_n are an iteration family for the operation $J_{K_1}(G)$ with $G_0 = K_1$.
- For forming the cliques K_n we need the operation of adding a vertex connected to all previous vertices.
  $J_{K_1}(G)$ is $MSOL$-smooth for $G = (V(G), E(G))$
  but not for $I(G) = (V(G) \cup E(G), R(G))$.

k-structures

Recall: A vocabulary τ is a set of relation symbols.

A τ-structure $\mathfrak{A}$ is an interpretation of the vocabulary τ over a **non-empty** universe A.

Definition 6 *For $k \in \mathbb{N}$, a k-τ-structure is a τ-structure with k additional unary relations $C_1^A, \ldots C_k^A$, called colors.*

We denote by τ_k the vocabulary $\tau \cup \{C_1, \ldots, C_k\}$.

Basic operations on k-τ-structures, I

Definition 7 (Adding a new colored element)

For $i \leq k$, the operation $Add_{C_i}(\mathfrak{A})$ adds a new element to A of color C_i.

More precisely, let $b \notin A$ and $\mathfrak{B} =< \{b\}, C_i^B = \{b\} >$.

Then $Add_{C_i}(\mathfrak{A}) = \mathfrak{A} \sqcup \mathfrak{B}$

Proposition 8 *$Add_{C_i}(\mathfrak{A})$ is $\mathbf{MSOL}$-smooth.*

Basic operations on k-τ-structures, II

Definition 9 (Recoloring)

For $i, j \leq k$, the operation $\rho_{i,j}(\mathfrak{A})$ recolors all elements of A of color i with color j.

More precisely, if the colors in $\mathfrak{A}$ are $C_1^A, \ldots, C_k^A$ then $\rho_{i,j}(\mathfrak{A}) = \mathfrak{B}$ has colors new colors $C_i^B = \emptyset$ and $C_j^B = C_i^A \cup C_j^A$ and all other colors and relations remain unchanged.

Proposition 10 *$\rho_{i,j}(\mathfrak{A})$ is $\mathbf{MSOL}$-smooth.*

Basic operations on k-τ-structures, III

Definition 11 (Adding tuples to relations)

The operation $\eta_{R,i_1,\ldots,i_m}(\mathfrak{A})$ *is defined as follows:*

For $R \in \tau$ *an* m*-ary relation symbol*
and for each $a_1 \in C_{i_1}^A, \ldots, a_m \in C_{i_m}^A$
we add the tuple $(a_1, \ldots, a_m)$ *to* R^A*.*

Proposition 12 $\eta_{i,j}(\mathfrak{A})$ *is* $\mathbf{MSOL}$*-smooth.*

Basic operations on k-τ-structures, IV

Definition 13 (Deleting tuples from relations)

The operation $\delta_{R,i_1,\ldots,i_m}(\mathfrak{A})$ is defined as follows:

For $R \in \tau$ an m-ary relation symbol
and for each $a_1 \in C_{i_1}^A, \ldots, a_m \in C_{i_m}^A$
we remove the tuple $(a_1, \ldots, a_m)$ from R^A.

Proposition 14 $\delta_{i,j}(\mathfrak{A})$ *is* $\mathbf{MSOL}$*-smooth.*

Basic operations on k-τ-structures, V

Definition 15 (Quantifierfree transductions)

For each $R \in \tau_k$ of arity $\alpha(R)$ let $\phi_R(x_1, \ldots, x_{\alpha(R)})$ be a **quantifierfree** *τ_k-formula with free first order variables as indicated.*

Let $\Phi =< \phi_R(x_1, \ldots, x_{\alpha(R)}) : R \in \tau_k >$.

The quantifier free transduction $\Phi^\star(\mathfrak{A})$ **redefines** *all the predicates R^A in $\mathfrak{A}$ by ϕ_R^A.*

Exercise:
All of the previous operations are special cases of Quantifierfree transductions.

Proposition 16 *Quantifier free transductions are $\mathbf{MSOL}$-smooth.*

$MSOL$-elementary and $MSOL$-smooth operations

Definition 17 *An operation F on τ_k-structures is* **$MSOL$-elementary** *if F is a finite composition of basic operations on τ_k-structures.*

Proposition 18 *Let F be $MSOL$-elementary and $\mathfrak{A}$ and $\mathfrak{B}$ two τ_k structures with $\mathfrak{A} \equiv_q^{MSOL} \mathfrak{B}$, then $F(\mathfrak{A}) \equiv_q^{MSOL} F(\mathfrak{B})$, hence, F is a $MSOL$-smooth.*

Proposition 19 *Let F be $MSOL$-elementary and $\mathcal{G}$ be an F-iteration family. Then $\mathcal{G}$ is of bounded clique-width.*

Corollary 20 *The families $I(K_n)$, $Grid_{n,n}$ are not of bounded clique-width. Hence they are* **not** *F-iteration families for any F which is $MSOL$-elementary.*

The Recurrence Theorem

E. Fischer and M. (2008)

Theorem 21 *Let*

- F *be an* $MSOL$*-smooth operation on* τ_k*-structures.*
- P *be a* τ*-polynomial which is* $MSOL(\tau)$*-definable.*
- $\mathcal{A} = \{A_n : n \in \mathbb{N}\}$ *be an* F*-iteration family of* τ*-structures.*

Then $\mathcal{A}$ *is* P*-recursive, i.e. there is* $q \in \mathbb{N}$*,*
and polynomials $p_1, \ldots, p_q \in \mathbb{Z}[\bar{X}]$ *such that for sufficiently large* n

$$P(G_{n+q+1}) = \sum_{i=1}^{q} p_i \cdot P(G_{n+i})$$

Proof ingredients

- For fixed q and a fixed number of free variables, there are, up to logical equivalence, only finitely many $MSOL(\tau)$-formulas of quantifier rank q.

  Let $\bar{P} = (\bar{P}_1, \ldots \bar{P}_\alpha)$ be the vector of all $MSOL(\tau)^q$-definable polynomials.

- Feferman-Vaught Theorem for $MSOL$-definable graph polynomials

  J.A. Makowsky, Algorithmic uses of the Feferman-Vaught Theorem,
  Annals of Pure and Applied Logic, 126 (2004), 159-213

- Bilinear version of the Feferman-Vaught Theorem for graph polynomials.

  With an $MSOL$-elementary operation F and a fixed q there is a matrix M_F such that

  $$\bar{P}(F(G)) = M_F \cdot \bar{P}(G)$$

- Use the characteristic polynomial of M_F.

Lecture 13 (Outline)

- More on transductions (with proofs)
- The Feferman-Vaught Theorem for **MSOL**-properties (with proofs)
- A Feferman-Vaught-like Theorem for **MSOL**-definable graph polynomials (with proofs)