Chinese Remainder Encoding for Hamiltonian Cycles

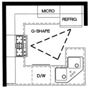
Marijn J.H. Heule

Carnegie Mellon University

SAT 2021

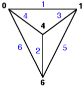
July 7, 2021

Encodings Matter

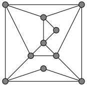


Architectural 3D Layout
[VSMM '07]
Henriette Bier

Edge-matching Puzzles
[LaSh '08]

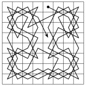


Graceful Graphs
[AAAI '10]
Toby Walsh

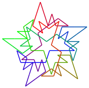


Clique-Width
[SAT '13, TOCL '15]
Stefan Szeider

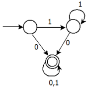
Firewall Verification
[SSS '16]
Mohamed Gouda



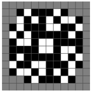
Open Knight Tours
Moshe Vardi



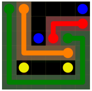
Van der Waerden numbers
[EJoC '07]



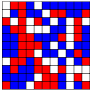
Software Model Synthesis
[ICGI '10, ESE '13]
Sicco Verwer



Conway's Game of Life
[EJoC '13]
Willem van der Poel



Connect the Pairs
Donald Knuth

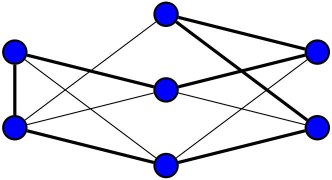


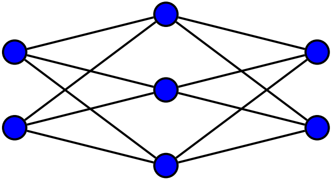
Pythagorean Triples
[SAT '16, CACM '17]
Victor Marek & Oliver Kullmann

Collatz conjecture [Open]
Emre Yolcu & Scott Aaronson
[CADE '21]

Hamiltonian Cycles: Two Constraints

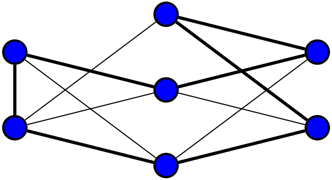
Hamiltonian Cycle Problem (HCP):
Does there exists a cycle that visits all vertices exactly once?

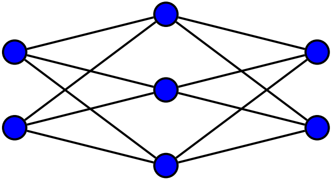




Hamiltonian Cycles: Two Constraints

Hamiltonian Cycle Problem (HCP):
Does there exists a cycle that visits all vertices exactly once?

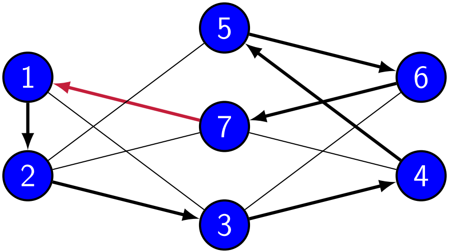




Two constraints:

- Exactly two edges per vertex: easy cardinality constraints
- Exactly one cycle: hard to be compact and arc-consistent
  - One option is to ignore the constraint: incremental SAT.
  - Various encodings use $O(|V|^3)$. Too large for many graphs.
  - For large graphs we need encodings that are quasi-linear in $|E|$.

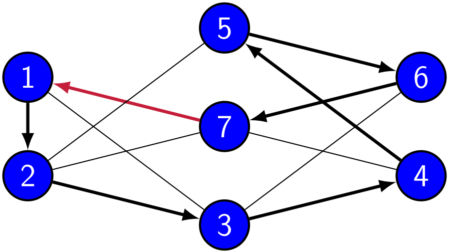
Hamiltonian Cycles: Encodings Quasi-Linear in |E|



Key elements:

- Each vertex have an index in the range $\{1, \ldots, |V|\}$.
- Selected edges are directed.
- Each vertex has one incoming and one outgoing edge.
- For each directed edge (u, v): the index of v is the successor of the index of u — except for the starting vertex.

Hamiltonian Cycles: Encodings Quasi-Linear in |E|



Key elements:

- Each vertex have an index in the range $\{1, \ldots, |V|\}$.
- Selected edges are directed.
- Each vertex has one incoming and one outgoing edge.
- For each directed edge (u, v): the index of v is the successor of the index of u — except for the starting vertex.

How to implement the successor property?

Hamiltonian Cycles: Binary Adder Encoding [Zhou 2020]

Each index is a binary number. If edge variable $e_{u,v}$ is assigned to true then the index of v is the successor of the index of u.

Example

Let $|V| = 7$, thus $k = \lceil \log_2 7 \rceil = 3$. For vertex v, variables v_2, v_4, and v_8 denote the least, middle, and most significant bit, respectively. For an edge variable $e_{u,v}$, we use the constraints:

$$
\begin{aligned}
e_{u,v} &\rightarrow (u_2 \not\leftrightarrow v_2) \\
(e_{u,v} \wedge \overline{u}_2) &\rightarrow (u_4 \leftrightarrow v_4) \\
(e_{u,v} \wedge u_2) &\rightarrow (u_4 \not\leftrightarrow v_4) \\
(e_{u,v} \wedge \overline{u}_2) &\rightarrow (u_8 \leftrightarrow v_8) \\
(e_{u,v} \wedge \overline{u}_4) &\rightarrow (u_8 \leftrightarrow v_8) \\
(e_{u,v} \wedge u_2 \wedge u_4) &\rightarrow (u_8 \not\leftrightarrow v_8)
\end{aligned}
$$

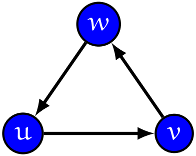
Hamiltonian Cycles: Binary Adder Encoding [Zhou 2020]

Each index is a binary number. If edge variable $e_{u,v}$ is assigned to true then the index of v is the successor of the index of u.

Example

Let $|V| = 7$, thus $k = \lceil \log_2 7 \rceil = 3$. For vertex v, variables v_2, v_4, and v_8 denote the least, middle, and most significant bit, respectively. For an edge variable $e_{u,v}$, we use the constraints:

$$\begin{aligned} e_{u,v} &\rightarrow (u_2 \nleftrightarrow v_2) \\ (e_{u,v} \wedge \overline{u}_2) &\rightarrow (u_4 \leftrightarrow v_4) \\ (e_{u,v} \wedge u_2) &\rightarrow (u_4 \nleftrightarrow v_4) \\ (e_{u,v} \wedge \overline{u}_2) &\rightarrow (u_8 \leftrightarrow v_8) \\ (e_{u,v} \wedge \overline{u}_4) &\rightarrow (u_8 \leftrightarrow v_8) \\ (e_{u,v} \wedge u_2 \wedge u_4) &\rightarrow (u_8 \nleftrightarrow v_8) \end{aligned}$$



$u_2 \rightarrow \neg v_2 \rightarrow w_2 \rightarrow \neg u_2$

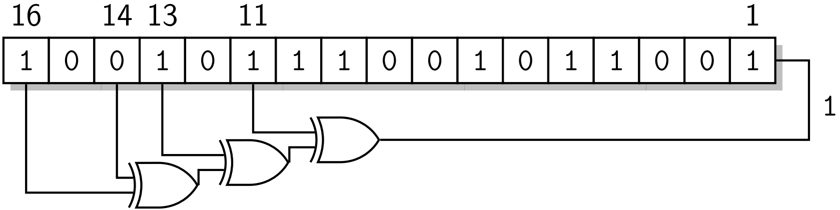
This encoding can quickly refute odd cycles

Hamiltonian Cycles: Linear-Feedback Shift Register

A k-bit Linear-Feedback Shift Register (LFSR) loops through $\{1, \ldots, 2^k - 1\}$ by shifting all bits one position to the left and placing the parity of some bits in the vacated position.

Example

An example LFSR of 16 bits is $x_{11} \oplus x_{13} \oplus x_{14} \oplus x_{16}$, which has $2^{16} - 1 = 65{,}535$ states. The figure below shows an illustration of this LFSR with state 1001011100101100 1. The next state is 0010111001011001 1.



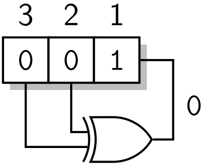
Hamiltonian Cycles: LFSR Encoding [Johnson 2018]

Enforcing the successor property using LFSR is compact and has been used to efficiently find Hamiltonian cycles in Erin and Stedman triples.

Example

Let $|V| = 7$, thus $k = \lceil \log_2(7+1) \rceil = 3$. We use 3-bit LFSR $x_2 \oplus x_3$. The bit-vector variables of vertex v are $v_{7,1}$, $v_{7,2}$, and $v_{7,3}$. For an edge variable $e_{u,v}$, we add the constraints:

$$\begin{aligned} e_{u,v} &\rightarrow (v_{7,1} \leftrightarrow (u_{7,2} \not\leftrightarrow u_{7,3})) \\ e_{u,v} &\rightarrow (v_{7,2} \leftrightarrow u_{7,1}) \\ e_{u,v} &\rightarrow (v_{7,3} \leftrightarrow u_{7,2}) \end{aligned}$$



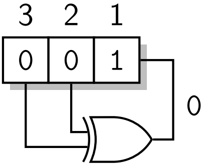
Hamiltonian Cycles: LFSR Encoding [Johnson 2018]

Enforcing the successor property using LFSR is compact and has been used to efficiently find Hamiltonian cycles in Erin and Stedman triples.

Example

Let $|V| = 7$, thus $k = \lceil \log_2(7+1) \rceil = 3$. We use 3-bit LFSR $x_2 \oplus x_3$. The bit-vector variables of vertex v are $v_{7,1}$, $v_{7,2}$, and $v_{7,3}$. For an edge variable $e_{u,v}$, we add the constraints:

$$
\begin{aligned}
e_{u,v} &\rightarrow (v_{7,1} \leftrightarrow (u_{7,2} \not\leftrightarrow u_{7,3})) \\
e_{u,v} &\rightarrow (v_{7,2} \leftrightarrow u_{7,1}) \\
e_{u,v} &\rightarrow (v_{7,3} \leftrightarrow u_{7,2})
\end{aligned}
$$



This encoding is compact and has lots of propagation

Hamiltonian Cycles: Chinese Remainder Encoding

Can we get the best all three worlds?

- Incremental SAT: Only partially encode the hard constraint
- Binary adder: refute some cycles quickly
- LSFR: few and short clauses, no auxiliary variables

Hamiltonian Cycles: Chinese Remainder Encoding

Can we get the best all three worlds?

- Incremental SAT: Only partially encode the hard constraint
- Binary adder: refute some cycles quickly
- LSFR: few and short clauses, no auxiliary variables

Chinese remainder encoding:

- Block all subcycles except one of length $0 \pmod{m}$
- Pick m (can be smaller than $|V|$) with small prime factors
- Enforce $0 \pmod{p_i}$ for each prime factor p_i of m
- Use LFSR for primes > 2 and binary adder for $p_i = 2$

Hamiltonian Cycles: Flinders HCP Challenge Graphs

Evaluation on reasonably large instances from the Flinders HCP Challenge Graphs suite

- Runtime (s) of CaDiCaL on binary adder and LFSR
- Smallest k such that 2^k (or $2^k - 1$) is larger than $|V|$

graph #	$\lvert V \rvert$	$\lvert E \rvert$	adder (2^k)	LSFR ($2^k - 1$)
424	2466	4240	> 3600	> 3600
446	2557	4368	> 3600	> 3600
470	2740	4509	2500.61	> 3600
491	2844	4267	173.46	245.92
506	2964	4447	78.29	244.48
522	3060	4591	84.51	611.46
526	3108	4663	160.73	544.97
529	3132	4699	69.69	275.13

Hamiltonian Cycles: Chinese Remainder Results

Evaluation with CaDiCaL on various cycle lengths (m)

- ✗ : First solution consists of multiple cycles
- ✓ : First solution consists of a single cycle

graph #	2	6	12	60	105	420
424	9.81 ✗	665.18 ✗	340.11 ✗	307.71 ✗	494.11 ✓	**488.70** ✓
446	13.24 ✗	334.62 ✗	169.52 ✗	380.47 ✗	**573.38** ✓	722.23 ✓
470	17.08 ✗	166.16 ✗	152.31 ✗	933.36 ✗	501.91 ✗	**840.89** ✓
491	0.06 ✗	22.04 ✗	**7.47** ✓	34.45 ✓	123.36 ✓	135.22 ✓
506	0.11 ✗	31.75 ✗	**19.24** ✓	33.48 ✓	28.73 ✓	63.20 ✓
522	0.63 ✗	5.66 ✗	32.95 ✓	133.40 ✓	**30.40** ✓	67.03 ✓
526	0.05 ✗	24.16 ✗	71.67 ✓	**34.37** ✓	34.69 ✗	158.69 ✓
529	0.40 ✗	17.90 ✗	60.19 ✓	48.09 ✓	**42.33** ✓	365.58 ✓

Conclusions and Future Work

Encodings matter

Chinese remainder encoding:

- Best of three worlds (partial, compact, refute short cycles)
- Block subcycles of length 0 modulo small primes
- Chinese remainder theorem: all cycles are of length 0 modulo the product of the primes

Future work:

- Use a similar encoding for other graph problems
- Explore the effectiveness for other solving techniques