A COMPARISON OF ALGORITHMS FOR PROVING POSITIVITY OF LINEARLY RECURRENT SEQUENCES

P. Nuspl, V. Pillwein

August 24, CASC 2022

JOHANNES KEPLER UNIVERSITY LINZ

Doctoral Program Computational Mathematics
Numerical Analysis and Symbolic Computation

Der Wissenschaftsfonds.

C-finite sequences

Definition

A sequence $c(n) \in \mathbb{Q}^{\mathbb{N}}$ is called C-finite if there are constants $\gamma_0, \ldots, \gamma_{r-1} \in \mathbb{Q}$ such that

$$c(n+r) = \gamma_0 c(n) + \cdots + \gamma_{r-1} c(n+r-1) \quad \text{for all } n \in \mathbb{N}.$$

Examples:

- Fibonacci numbers,
- Pell numbers,
- Perrin numbers.

Problem

Problem

Does $c(n) > 0$ hold for all $n \in \mathbb{N}$?

- In general, it is not known whether the problem is decidable.
- For examples appearing in practice, it usually is decidable (as we will see).
- Which algorithms can be used to prove positivity?

Example (A007910)

Consider the rational function

$$\frac{1}{(1-2x)(1+x^2)} = \sum_{n\geq 0} c(n)x^n.$$

The coefficient sequence $c(n)$ is C-finite satisfying

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

Are all coefficients positive, i.e., $c(n) > 0$ for all $n \in \mathbb{N}$?

Theorem (folklore)

A sequence $c(n)$ is C-finite if and only if the generating function $\sum_{n\geq 0} c(n)x^n$ is a rational function.

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- We try to show positivity by induction:

$$(c(n) > 0 \land c(n+1) > 0 \land c(n+2) > 0)$$
$$\implies c(n+3) > 0.$$

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- We try to show positivity by induction:

$$\begin{aligned}&(c(n) > 0 \wedge c(n+1) > 0 \wedge c(n+2) > 0)\\ \implies &\; 2c(n) - c(n+1) + 2c(n+2) > 0.\end{aligned}$$

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- We try to show positivity by induction:

$$\begin{aligned}(c(n) > 0 \land c(n+1) > 0 \land c(n+2) > 0)\\ \implies 2c(n) - c(n+1) + 2c(n+2) > 0.\end{aligned}$$

- Let's translate this to formula which can be verified automatically:

$$\forall y_0, y_1, y_2 \in \mathbb{R}\colon\ (y_0 > 0 \land y_1 > 0 \land y_2 > 0) \implies 2y_0 - y_1 + 2y_2 > 0.$$

  Quantifier elimination yields `False`.

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- We try to show positivity by induction:

$$\begin{aligned}&(c(n) > 0 \land c(n+1) > 0 \land c(n+2) > 0)\\ &\implies 2c(n) - c(n+1) + 2c(n+2) > 0.\end{aligned}$$

- Let's translate this to formula which can be verified automatically:

$$\forall y_0, y_1, y_2 \in \mathbb{R}\colon\ (y_0 > 0 \land y_1 > 0 \land y_2 > 0) \implies 2y_0 - y_1 + 2y_2 > 0.$$

  Quantifier elimination yields `False`.
- Neither proves nor disproves that sequence is positive.

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- Let's iterate the induction formula:

$$(c(n) > 0 \land c(n+1) > 0 \land c(n+2) > 0 \land c(n+3) > 0)$$
$$\implies c(n+4) > 0.$$

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- Let's iterate the induction formula:

$$(c(n) > 0 \wedge c(n+1) > 0 \wedge c(n+2) > 0 \wedge 2c(n) - c(n+1) + 2c(n+2) > 0)$$
$$\implies 4c(n) + 3c(n+2) > 0.$$

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- Let's iterate the induction formula:

$$(c(n) > 0 \wedge c(n+1) > 0 \wedge c(n+2) > 0 \wedge 2c(n) - c(n+1) + 2c(n+2) > 0)$$
$$\implies 4c(n) + 3c(n+2) > 0.$$

- The new input for quantifier elimination therefore reads as:

$$\forall y_0, y_1, y_2 \in \mathbb{R}\colon\ (y_0 > 0 \wedge y_1 > 0 \wedge y_2 > 0 \wedge 2y_0 - y_1 + 2y_2 > 0)$$
$$\implies 4y_0 + 3y_2 > 0.$$

Quantifier elimination yields `True`.

Gerhold-Kauers method: Example

We have

$$c(n+3) = 2c(n) - c(n+1) + 2c(n+2), \quad c(0) = 1, c(1) = 2, c(2) = 3.$$

- Let's iterate the induction formula:

$$(c(n) > 0 \wedge c(n+1) > 0 \wedge c(n+2) > 0 \wedge 2c(n) - c(n+1) + 2c(n+2) > 0)$$
$$\implies 4c(n) + 3c(n+2) > 0.$$

- The new input for quantifier elimination therefore reads as:

$$\forall y_0, y_1, y_2 \in \mathbb{R}\colon\ (y_0 > 0 \wedge y_1 > 0 \wedge y_2 > 0 \wedge 2y_0 - y_1 + 2y_2 > 0)$$
$$\implies 4y_0 + 3y_2 > 0.$$

  Quantifier elimination yields `True`.
- Checking $c(0), \dots, c(3) > 0$ proves that $c(n) > 0$ for all $n \in \mathbb{N}$

Gerhold-Kauers method

- This is known as the Gerhold-Kauers method (Gerhold and Kauers 2005).
- It is not guaranteed to work:
  - If the sequence is not positive, the algorithm will find a counterexample.
  - If the sequence is positive, the algorithm might not terminate (some conditions for termination are known: e.g., Kauers and Pillwein 2010).
- It can be used for other sequences as well (e.g., P-recursive sequences).
- There are variations which can be more powerful.

Closed form

Theorem (folklore)

Let $c(n)$ be C-finite. Then, there is an $n_0 \in \mathbb{N}$ and polynomials $p_1, \dots, p_m \in \overline{\mathbb{Q}}[x]$ and constants $\lambda_1, \dots, \lambda_m \in \overline{\mathbb{Q}}$ such that

$$c(n) = \sum_{i=1}^{m} p_i(n)\lambda_i^n \quad \text{for all } n \geq n_0.$$

We call the λ_i the eigenvalues of the sequence c.

In our example we have

$$c(n) = \tfrac{4}{5}\, 2^n + \left(\tfrac{1}{10} - \tfrac{1}{5}i\right) i^n + \left(\tfrac{1}{10} + \tfrac{1}{5}i\right)(-i)^n \quad \text{for all } n \in \mathbb{N},$$

so the sequence has the eigenvalues $2, i, -i$. Clearly, the sequence will be positive eventually.

Analytic method

We want to show positivity of

$$c(n) = \tfrac{4}{5}\,2^n + \underbrace{\left(\tfrac{1}{10} - \tfrac{1}{5}i\right) i^n + \left(\tfrac{1}{10} + \tfrac{1}{5}i\right) (-i)^n}_{=:r(n)} = \tfrac{4}{5}\,2^n + r(n).$$

Clearly

$$|r(n)| \le |\tfrac{1}{10} - \tfrac{1}{5}i||i|^n + |\tfrac{1}{10} + \tfrac{1}{5}i||-i|^n = \tfrac{1}{\sqrt{5}}.$$

Hence,

$$c(n) = \tfrac{4}{5}\,2^n + r(n) \ge \tfrac{4}{5}\,2^n - |r(n)| = \tfrac{4}{5}\,2^n - \tfrac{1}{\sqrt{5}} > 0$$

for all $n \in \mathbb{N}$, so $c(n)$ is positive.

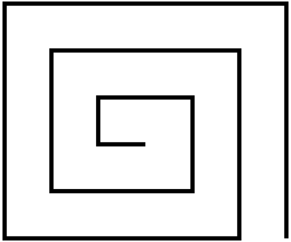
Analytic method

- This method always works if there is a unique dominant eigenvalue, i.e., we have
$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_m|.$$
- Can easily be implemented using arbitrary precision arithmetic or algebraic number arithmetic.
- Analytic method can be extened for sequences with at most 5 dominant eigenvalues (Ouaknine and Worrell 2014).
- For sequences with more than 5 dominant eigenvalues, it is not known whether checking positivity is decidable.

Example 2 (A000969)

Consider the sequence

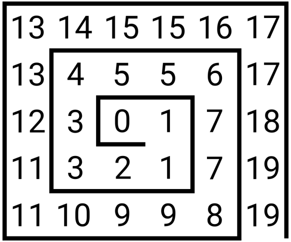
$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$



Example 2 (A000969)

Consider the sequence

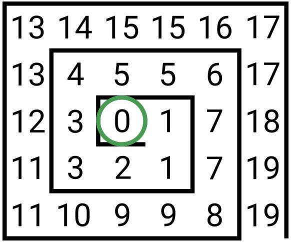
$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$



Example 2 (A000969)

Consider the sequence

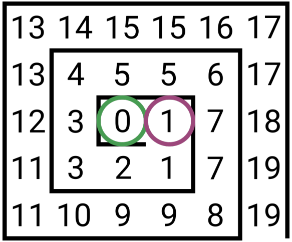
$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$



Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

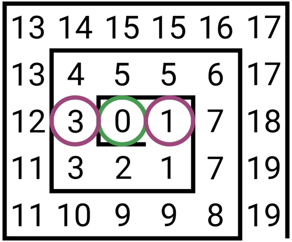


$$1$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

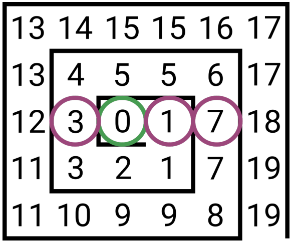


$$1, 3$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

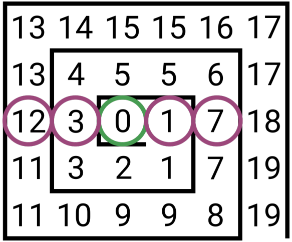


$$1, 3, 7$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

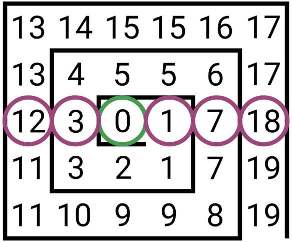


$$1, 3, 7, 12$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

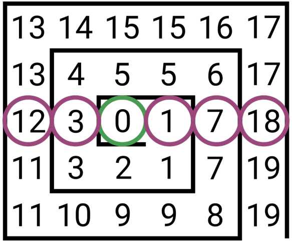


$$1, 3, 7, 12, 18$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$

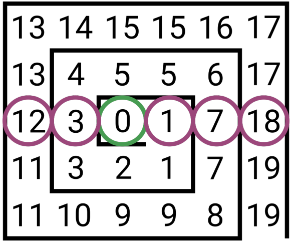


$$1, 3, 7, 12, 18, 26, 35, 45, 57, 70, 84, 100, 117, \ldots$$

Example 2 (A000969)

Consider the sequence

$$0, 1, 1, 2, 3, 3, 4, 5, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, \ldots$$



$$1, 3, 7, 12, 18, 26, 35, 45, 57, 70, 84, 100, 117, \ldots$$

This sequence is C-finite satisfying

$$c(n+5) = c(n) - 2c(n+1) + c(n+2) - c(n+3) + 2c(n+4).$$

Decomposition

We have

$$c(n+5) = c(n) - 2c(n+1) + c(n+2) - c(n+3) + 2c(n+4).$$

- The sequence has the eigenvalues $1, \frac{-1\pm\sqrt{3}\,i}{2}$, the latter being roots of unity.
- Neither the Gerhold-Kauers method nor the analytic method works.
- The subsequences $c(3n), c(3n+1), c(3n+2)$ all have a unique dominant root and we can therefore easily show positivity of all three.
- This gives rise to the positivity of c.
- There is no guarantee that such a decomposition can be found, but usually it works.

Experiments

- We implemented these algorithms (and more) in SageMath and Mathematica.
- Tested them on 1000 positive C-finite sequences from the OEIS with orders

order	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	> 15
	73	134	117	139	120	80	87	36	47	27	31	14	17	10	10	58

- For how many could the SageMath implementation prove positivity with a 60 seconds timeout?

method	Gerhold-Kauers	Analytic	Decomposition
# successfully proven	384	566	984

- Given more time, each of the 1000 sequences could be proven to be positive.

Package: SageMath

Our SageMath package `rec_sequences` provides several methods to prove positivity of C-finite sequences[1]:

```
sage: from rec_sequences.CFiniteSequenceRing import *
sage: C = CFiniteSequenceRing(QQ)
sage: c1 = C([2,-1,2,-1], [1,2,3])
sage: c1 > 0
True
sage: c2 = C([1,-2,1,-1,2,-1], [1,3,7,12,18])
sage: c2 > 0
True
```

[1] It is available at https://github.com/PhilippNuspl/rec_sequences.

Package: Mathematica

For Mathematica our package `PositiveSequence` can be used to prove positivity of C-finite sequences[2]:

In[1]:= $\ll$ **RISC`PositiveSequence`**

In[2]:= $\mathbf{c1 = RE[\{\{0, 2, -1, 2, -1\}, \{1, 2, 3\}\}, c[n]];}$

In[3]:= $\mathbf{PositiveSequence}[\boldsymbol{c}\mathbf{1}]$

Out[3]= True

In[4]:= $\mathbf{c2 = RE[\{\{0, 1, -2, 1, -1, 2, -1\}, \{1, 3, 7, 12, 18\}\}, c[n]];}$

In[5]:= $\mathbf{PositiveSequence}[\boldsymbol{c}\mathbf{2}]$

Out[5]= True

[2] It is available at https://www.risc.jku.at/research/combinat/software/PositiveSequence/.

Conclusions

What have we done?

- Compared several well known and new methods for automatically proving positivity of C-finite sequences.
- Basic methods already cover most sequences encountered in practice.
- Provide implementations in SageMath and Mathematica.

What is left?

- Other, more sophisticated methods are known:
  - Are they more efficient?
  - Do they cover more sequences that appear in practice?
- Methods for P-recursive sequences, i.e., sequences satisfying a linear recurrence with polynomial coefficients.

References

[1] Stefan Gerhold and Manuel Kauers. "A Procedure for Proving Special Function Inequalities Involving a Discrete Parameter". In: Proceedings of ISSAC 2005, Beijing, China, July 24–27, 2005. 2005, pp. 156–162.

[2] Manuel Kauers and Veronika Pillwein. "When Can We Detect That a P-Finite Sequence is Positive?". In: Proceedings of ISSAC 2010, Munich, Germany. New York, NY, USA: Association for Computing Machinery, 2010, pp. 195–201.

[3] Joël Ouaknine and James Worrell. "Positivity problems for low-order linear recurrence sequences". In: SODA '14: Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms. 2014, pp. 366–379.