A Procedure for Proving Special Function Inequalities Involving a Discrete Parameter

Stefan Gerhold[*]
RISC-Linz
Johannes Kepler Universität
A-4040 Linz, Austria
sgerhold@risc.uni-linz.ac.at

Manuel Kauers[†]
RISC-Linz
Johannes Kepler Universität
A-4040 Linz, Austria
mkauers@risc.uni-linz.ac.at

ABSTRACT

We define a class of special function inequalities that contains many classical examples, such as the Cauchy-Schwarz inequality, and introduce a proving procedure based on induction and Cylindrical Algebraic Decomposition. We present an array of non-trivial examples that can be done by our method. Most of them have not been proven automatically before. Some difficult well-known inequalities such as the Askey-Gasper inequality and Vietoris's inequality lie in our class as well, but we do not know if our proving procedure terminates for them.

Categories and Subject Descriptors

I.1.2 [**Computing Methodologies**]: Symbolic and Algebraic Manipulation—*Algorithms*; G.2.1 [**Discrete Mathematics**]: Combinatorics—*Recurrences and difference equations*

General Terms

Algorithms

Keywords

Inequalities, Special functions, Computer Proofs, Cylindrical Algebraic Decomposition

1. INTRODUCTION

While there is a panoply of algorithms for proving combinatorial and special function identities, not much is known concerning the algorithmic treatment of inequalities. The importance of studying inequalities hardly needs to be emphasized. For instance, virtually any proof in analysis contains an estimation of some sort. There are classical textbooks [12, 15, 16, 23] on this broad subject, but, of course, the possibility of proving a given inequality automatically has some obvious advantages in comparison to table lookup.

For some classical inequalities there is an underlying identity that makes the truth of the inequality obvious. For instance, Lagrange's identity

$$\sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2 - \Bigl(\sum_{k=1}^n x_k y_k\Bigr)^2 = \sum_{1\le k<i\le n} (x_k y_i - x_i y_k)^2$$

immediately implies the Cauchy-Schwarz inequality

$$\Bigl(\sum_{k=1}^n x_k y_k\Bigr)^2 \le \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2 \quad (n \ge 0). \tag{1}$$

Askey and Gasper [2] found an identity that establishes the inequality

$$\sum_{k=0}^n P_k^{(\alpha,0)}(x) > 0 \qquad (\alpha > -1, -1 < x \le 1) \tag{2}$$

for a sum of Jacobi polynomials, which was applied in the first proof of the Bieberbach conjecture [9]. Regardless of whether such an identity is algorithmically provable [14, 10], finding a suitable one by human insight is indispensable for this line of attack.

The inequality

$$\sum_{k=n}^\infty \frac{1}{k^2\binom{n+k}{k}} < \frac{2}{n\binom{2n}{n}} \qquad (n \ge 1) \tag{3}$$

arose in work by Knopp and Schur [22]. Paule's proof [19] of (3) contains an application of the extended Gosper algorithm [21]. Also, Paule [18] has applied several computer algebra tools in a proof of another inequality, which implies a conjecture of Knuth. Although computer algebra assisted, both proofs are altogether by no means mechanical.

For some inequalities, computer algebra systems are able to simplify the functions that occur in it and verify the resulting inequality automatically, but this functionalities are rather limited.

The present paper is an attempt at proving inequalities entirely automatically. All inequalities under consideration must depend on a discrete parameter, such as n in (1), (2)

[*] Supported by the Austrian Science Foundation (FWF) grant F1305

[†] Partially supported by the Austrian Science Foundation (FWF) grant F1305 and the German Academic Exchange Service (DAAD) grant D/03/40515

ISSAC'05, July 24–27, 2005, Beijing, China.

and (3). Our method is based on using Cylindrical Algebraic Decomposition (CAD) for constructing an induction proof for the conjectured inequality, given recurrence relations defining the involved quantities as specified in Section 2. The proving procedure, presented in Section 3, is not backed by a termination theorem, but it works surprisingly well on concrete examples. Section 4 is devoted to a collection of nontrivial examples that we were able to tackle with our method, and in Section 5 we show how our procedure can be applied to the problem of analyzing the sign pattern of oscillating sequences.

2. PRELIMINARIES

The inequalities in our consideration involve sequences that can be expressed by means of difference fields. A short introduction to the theory of difference fields, from which we need hardly more than the basic definitions, is given in this section. Later, we will introduce the set of formulas to which our method can be applied. These are essentially boolean combinations of inequalities on both sides of which there are sequences which can be expressed in terms of difference fields.

2.1 Difference Fields and Sequences

A field K together with a distinguished endomorphism $s\colon K \to K$ is called a *difference field.* The map s is called the *shift operator* of the difference field. The theory of difference fields [6] allows an algebraic treatment of difference equations and sequences, quite in analogy to the treatment of differential equations and analytic functions by differential algebra.

If $K = (K, s)$ is a difference field, then for any $r \in \mathbb{N}$ and any rat $\in K' := K(t^{(0)}, t^{(1)}, \dots, t^{(r-1)})$, we can turn K' into a difference field by extending s to $s'\colon K' \to K'$ via

$$\begin{aligned} s'(t^{(i)}) &:= t^{(i+1)} & i = 0, \dots, r-2,\\ s'(t^{(r-1)}) &:= \operatorname{rat}(t^{(0)}, t^{(1)}, \dots, t^{(r-1)}). \end{aligned}$$

The objects $t^{(0)}, t^{(1)}, \dots, t^{(r-1)}$ are understood as algebraically independent indeterminates over K, and we will often write $t, st, \dots, s^{r-1}t$ for them. We say that K' is obtained from K by *adjoining a difference variable t to K.* The number r is called the *order* of t. We write $K' := K\{t\}$ and use the same symbol, s, for s and s'. Difference fields can be constructed by starting with a field C of constants (i.e., $s(c) = c$ for all $c \in C$) and successively adjoining difference variables $t_1, t_2, \dots, t_m$ as above. The resulting difference field is denoted by $C\{t_1, \dots, t_m\}$. More general constructions are possible [6], but not of interest in this paper.

A difference variable may serve as an algebraic formalization of a sequence. Let $K = C\{t_1, \dots, t_m\}$ be a difference field and $r_1, \dots, r_m$ be the respective orders of $t_1, \dots, t_m$. Then K uniquely defines sequences $[t_1], \dots, [t_m]$ over C as soon as initial values $[t_i]_1, \dots, [t_i]_{r_i} \in C$ are chosen for each i. The value $[t_i]_n$ is defined recursively via $[t_i]_{n+1} = [st_i]_n$. For bad choices of initial values, the sequences may not be well-defined, because iterated application of the recurrence may eventually lead to a division by zero.

The notation $[\cdot]$ is not only used for the sequences induced by the difference variables $t_1, \dots, t_m$, but for any element of the difference field. Observe that the (possibly partial) mapping $[\cdot]\colon C\{t_1, \dots, t_m\} \to C^{\mathbb{N}}$ only depends on the choice of some finitely many initial values.

It is possible to define quite a few interesting sequences by means of difference fields, including, e.g., all holonomic sequences. Many examples and useful closure properties for the class of expressible sequences are given in [13, 14].

2.2 Inequalities and Formulas

We consider formulas involving inequalities for sequences over $C := \mathbb{R} \cap \bar{\mathbb{Q}}$, the field of real algebraic numbers. Suppose a difference field $K = C\{t_1, \dots, t_m\}$ and a suitable choice of initial values is given. An *atomic formula* over K is of the form $k_1 \diamondsuit k_2$ with $k_1, k_2 \in K$ and $\diamondsuit \in \{=, \neq, \leq, \geq, <, >\}$. A *formula* over K is a boolean combination of true, false, and atomic formulas over K. For example,

$$\phi = \; t_1 > 0 \land \neg(s^2 t_1 < 1 \lor (st_2 + 1)^3 \leq t_2)$$

is a formula over $C\{t_1, t_2\}$. The definition of s is extended to formulas in the obvious way.

Formulas over K give rise to formulas for sequences. The meaning of square brackets introduced in the previous section is extended to formulas. Continuing the example, we have

$$[\phi]_n = \; [t_1]_n > 0 \land \neg([t_1]_{n+2} < 1 \lor ([t_2]_{n+1} + 1)^3 \leq [t_2]_n).$$

If ϕ is a formula then $[\phi]$ is a sequence of truth values. We say that ϕ is *valid* if $[\phi]_1, [\phi]_2, \dots$ are all true. Note that for any specific number n, the truth of $[\phi]_n$ can easily checked by inspection.

3. THE PROVING PROCEDURE

Similarly to algorithms for proving identities, which reduce questions about recurrences to questions about algebraic sets that can be answered by Gröbner basis techniques, we will reduce the question whether a formula ϕ is valid to questions about semi-algebraic sets which can be answered by Cylindrical Algebraic Decomposition [7, 5].

As a subroutine for the proving procedure given below, we will use a function Reduce which transforms a formula ϕ over $C\{t_1, \dots, t_m\}$ into an equivalent formula $\phi' := \operatorname{Reduce}(\phi)$ such that $\phi' =$ false whenever ϕ is unsatisfiable. Here we regard $C\{t_1, \dots, t_m\}$ just as a field of rational functions over C, ignoring its difference field structure, and equivalence and satisfiability are meant in this sense: the formula $\phi' \iff \phi$ is true if it is true for any choice of real numbers in place of the $s^i t_j$, and ϕ is called unsatisfiable if it is not true for any choice of real numbers in place of the $s^i t_j$. For sets of formulas, we define

$$\operatorname{Reduce}(\{\phi_1, \dots, \phi_n\}) := \operatorname{Reduce}(\phi_1 \land \dots \land \phi_n).$$

Evidently, a function Reduce can be implemented by a standard application of CAD.

The basic idea of our procedure for proving a formula ϕ for sequences defined by difference fields is as follows: Like in the algorithm introduced in [13, 14] for identities, we want to construct a proof by complete induction on n. To this end, we first choose a number r_0 and try to refute

$$\psi := \big(\phi \land \dots \land s^{r_0 - 1}(\phi)\big) \land \neg s^{r_0}(\phi)$$

by Reduce. If ψ is indeed inconsistent, we have obtained the induction step for proving the desired inequality. If the refutation does not succeed, we choose another number r_1

and try to refute

$$\psi' := \big(\phi \wedge \cdots \wedge s^{r_0-1}(\phi)\big) \wedge \big(\neg\psi \wedge \cdots \wedge \neg s^{r_1-1}(\psi)\big) \wedge s^{r_1}(\psi),$$

and, if necessary, iterate. The input of the procedure may include a knowledge base Φ of known valid formulas.

INPUT:
– a difference field $C\{t_1, \ldots, t_m\}$ and initial values for the sequences $[t_1], \ldots, [t_m]$
– a formula ϕ over $C\{t_1, \ldots, t_m\}$ to be proven
– a set Φ of formulas over $C\{t_1, \ldots, t_m\}$ s.t. ψ is valid for all $\psi \in \Phi$ ($\Phi = \emptyset$ allowed)
OUTPUT:
– true if ϕ is valid, false otherwise.

```
1. $i := 0$
2. Choose a number $r_0 \in \mathbb{N}$
3. If $[\phi]_n$ is false for some $n < r_0$, return false
4. $\Theta := \Phi \cup \{\phi, \ldots, s^{r_0-1}(\phi)\}$
   $\psi_0 :=$ Reduce($\Theta \cup \{\neg s^{r_0}(\phi)\}$)
5. While $\psi_i \neq$ false repeat
       $i := i + 1$
       Choose a number $r_i \in \mathbb{N}$
       If $[\psi_{i-1}]_n$ is true for some $n < r_i$, return false
       $\Theta := \Theta \cup \{\neg\psi_{i-1}, \ldots, \neg s^{r_i-1}(\psi_{i-1})\}$
       $\psi_i :=$ Reduce($\Theta \cup \{s^{r_i}(\psi_{i-1})\}$)
6. Return true
```

Correctness of the procedure is ensured by the following theorem.

THEOREM 1. *If the procedure above returns* true*, then* ϕ *is valid. If it returns* false*, then* $[\phi]_n =$ false *for at least one* $n \in \mathbb{N}_0$*.*

PROOF. Suppose the procedure returns true after ℓ iterations of the while loop, i.e., $\psi_\ell =$ false. We show by induction on n that $[\phi]_n$ is true for all $n \geq 0$ and that $[\psi_k]_n$ is false for all $n \geq 0$ ($k = 0, \ldots, \ell - 1$).

W.l.o.g. we may assume $\Phi = \emptyset$. For convenience of notation define $\chi_0 := \phi$ and $\chi_k := \neg\psi_{k-1}$ for $k = 1, \ldots, \ell + 1$. Step 3 and line 3 of step 5 assert that $[\chi_k]_n$ is true for $n = 0, \ldots, r_k - 1$. This serves as induction base. As for the induction step, we show that

$$\bigwedge_{i=0}^{\ell} \bigwedge_{j=0}^{r_i-1} s^j(\chi_i) \Longrightarrow \bigwedge_{i=0}^{\ell} s^{r_i}(\chi_i). \tag{4}$$

According to the definitions of χ_i and ψ_i and the specification of Reduce we have

$$\begin{aligned}
\chi_k = \neg\psi_{k-1} &\iff \neg\Big(\bigwedge_{i=0}^{k-1} \bigwedge_{j=0}^{r_i-1} s^j(\chi_i) \wedge \neg s^{r_{k-1}}(\chi_{k-1})\Big) \\
&\iff \neg \bigwedge_{i=0}^{k-1} \bigwedge_{j=0}^{r_i-1} s^j(\chi_i) \vee s^{r_{k-1}}(\chi_{k-1}) \\
&\iff \Big(\bigwedge_{i=0}^{k-1} \bigwedge_{j=0}^{r_i-1} s^j(\chi_i) \Longrightarrow s^{r_{k-1}}(\chi_{k-1})\Big) \qquad (5)
\end{aligned}$$

for $k = 1, \ldots, \ell + 1$. If the left hand side of (4) is true, then in particular each χ_k and each

$$\bigwedge_{i=0}^{k-1} \bigwedge_{j=0}^{r_i-1} s^j(\chi_i)$$

is true ($k = 1, \ldots, \ell$), so by (5) each $s^{r_{k-1}}(\chi_{k-1})$ is true ($k = 1, \ldots, \ell$). Furthermore, $\chi_{\ell+1} = \neg\psi_\ell =$ false by the termination criterion of the while loop, hence, again by (5) and the left hand side of (4), also $s^{r_\ell}(\chi_\ell)$ is true. Altogether we have shown that the right hand side of (4) is true. This establishes correctness if the procedure returns true.

Now consider the case where the procedure returns false. Then $[\chi_k]_n =$ false for some k and some n. Repeated usage of (5) shows that then $[\chi_0]_{r_0+r_1+\cdots+r_k} =$ false, and hence ϕ is not valid. □

Of course the method can be easily adapted to prove the validity of formulas ϕ for $n \geq n_0$ instead of $n \geq 0$ for any given $n_0 \in \mathbb{Z}$. Furthermore, it is also possible to allow continuous parameters x in the formulas, and constraints can also be specified for those parameters.

Unfortunately there is no guarantee that the procedure will terminate. In fact, there are examples where it does not. The simplest example where nontermination occurs might be the inequality $3^n > 2^n$: Define $s(t_1) = 2t_1, s(t_2) = 3t_2$ and consider $\phi := t_1 < t_2$. Then an easy induction shows

$$\psi_i \iff t_1 < 0 \wedge (\tfrac{2}{3})^{r_0+\cdots+r_{i-1}} t_1 < t_2 \leq (\tfrac{2}{3})^{r_0+\cdots+r_i} t_1,$$

and this is consistent for any choice of i and $r_0, \ldots, r_i$.

In order to get a proof for $3^n > 2^n$, it is necessary to supply additional information. For example, termination can be obtained by adding $2^n > 0$ or $3^n > 0$ to the knowledge base Φ. This is typical: for most of the examples given in Section 4.1 below, we had to supply additional information as well. However, it is usually sufficient to supply only trivial constraints such as $n > 0$ etc. to make the proof go through. It is also worth noting that additional information, which was supplied for a proof, often can itself be proven correct by the same procedure, if desired.

4. EXAMPLES

This section contains a collection of examples. We begin with a detailed execution of the proving procedure for proving the simple inequality

$$n! \sum_{k=0}^{n} \frac{(-1)^k}{k!} > 0 \qquad (n \geq 2),$$

which asserts the positivity of the derangement numbers. A suitable difference field can be constructed from the subexpressions of the formula. Take $C\{t_1, t_2, t_3, t_4\}$ with

$$\begin{aligned}
s(t_1) &= t_1 + 1 && (\, t_1 \sim n \,) \\
s(t_2) &= (t_1 + 1)t_2 && (\, t_2 \sim n! \,) \\
s(t_3) &= -t_3 && (\, t_3 \sim (-1)^n \,) \\
s(t_4) &= t_4 - \frac{t_3}{(t_1 + 1)t_2} && (\, t_4 \sim \Sigma \,)
\end{aligned}$$

The construction of these recurrences is the same as in [14]. Let us assume $\Phi = \{t_1 > 0\}$ is the set of known facts. We have to apply the procedure to $\phi = t_2 t_4 > 0$.

We choose $r_0 := 1$ and have to check one initial value. As the inequality is claimed only for $n \geq 2$, we check that $[\phi]_2$

is true (cf. the remark right after the proof of Theorem 1): $2(1-1+\frac{1}{2}) = 1 > 0$. We set

$$\psi_0 := \mathrm{Reduce}(\Phi \cup \{t_2 t_4 > 0, s(t_2 t_4) \le 0\}).$$

A suitable choice for Reduce is Mathematica's command CylindricalDecomposition [24], which returns

$$\psi_0 = t_1 > 0 \land \big(t_2 < 0 \land t_3 < 0 \land \frac{t_3}{(1+t_1)t_2} \le t_4 < 0$$
$$\lor\, t_2 > 0 \land t_3 > 0 \land 0 < t_4 \le \frac{t_3}{(1+t_1)t_2}\big).$$

This is not the boolean false, so we have to continue the while loop. We choose $r_1 = 1$ and check the initial value $[\psi_0]_2 = \text{false}$. Next we compute

$$\psi_1 := \mathrm{Reduce}(\Phi \cup \{t_4 > 0, \neg\psi_0, s(\psi_0)\}).$$

Again using Mathematica's CylindricalDecomposition, we now obtain $\psi_1 = \text{false}$, hence the proof is complete.

The choice of the values r_i in the procedure is arbitrary. Different choices lead to the same result, but may differ with respect to efficiency. If we had chosen $r_0 = 2$ instead of $r_0 = 1$, we would have found $\psi_0 = \text{false}$ and an iteration would not have been necessary. A reasonable choice for most examples is $r_i = 1$.

4.1 Example Gallery

Our method can be successfully applied to a number of nontrivial problems from textbooks on inequalities [12, 15, 16]. We were able to prove, among others, the following examples by our method, using only trivial facts like $n > 0$ as additional knowledge.

To our knowledge, none of these inequalities can been proven by any other symbolic computation method.

- 3.27 in [15]:
$$\frac{1}{4n} < \frac{1}{16^n}\binom{2n}{n}^2 < \frac{1}{3n+1} \qquad (n \ge 2)$$
- Levin's inequality (3.2.12 in [16]):
$$1 \le \frac{1+nx^{n+1}}{(1+n)x^n} \le 1 + \frac{1}{2}n(1-x)^2 x^{-n}$$
$(0 < x \le 1, n \ge 0)$
- 3.3.38 in [16]: If $F_n(x)$ denotes the nth Fibonacci polynomial, defined by
$$F_{n+2}(x) = xF_{n+1}(x) + F_n(x), \quad F_1(x) = 1, F_2(x) = x,$$
then
$$F_n(x)^2 \le (x^2-1)^2(x^2+2)^{n-3} \qquad (n \ge 3, x \in \mathbb{R})$$

There is no need to restrict the defining recurrences to rational functions. Inequalities involving algebraic functions can be treated as well, provided that the algebraic functions are simple enough so that the CAD implementation can deal with them. This makes it possible to automatize the proofs for the following inequalities:

- [17] If (R_n) is defined by
$$R_{n+1} = 1 + \frac{n}{R_n} \quad (n \ge 1), \qquad R_1 = 1,$$
then
$$\sqrt{n - \tfrac{3}{4}} \le R_n - \tfrac{1}{2} \le \sqrt{n + \tfrac{1}{4}} \qquad (n \ge 1)$$
- A variation of 4.15 in [15]:
$$\Big(\sum_{k=1}^{n} \sqrt{k}\Big)^2 \le \Big(\sum_{k=1}^{n} \sqrt[3]{k}\Big)^3 \qquad (n \ge 1)$$
- 11.1 in [15]:
$$\sqrt{n + \sqrt{(n-1) + \sqrt{\cdots + \sqrt{2 + \sqrt{1}}}}} < \sqrt{n} + 1$$
$(n \ge 1)$

Another bunch of interesting examples makes use of the idea that was introduced in [14], by which it is possible to bring sequences of variables $(x_n)_{n\ge1}$ into the scope of our approach. The idea is to regard $x_n, x_{n+1}, x_{n+2}, \ldots$ as a sequence without defining recurrence, and to introduce a fresh variable upon each shift. See [14] for details. Using this idea, it is possible to prove the Cauchy-Schwarz inequality entirely automatically without resorting to an underlying identity as outlined in the introduction. The following inequalities become completely routine, too.

- Thm. 6 on p. 112 in [16]:
$$\sum_{k=1}^{n}(-1)^{k-1}a_k^2 \ge \Big(\sum_{k=1}^{n}(-1)^{k-1}a_k\Big)^2$$
if $(a_k)_{k\ge1}$ is positive and decreasing
- Variations of 7.31 and 7.32 from [15]:
$$\sum_{k=2}^{n}\frac{\sum_{i=1}^{k} x_i}{\sum_{i=1}^{k-1} x_i} > \frac{n^2}{n-1}, \qquad \sum_{k=2}^{n}\frac{x_k}{\sum_{i=1}^{k-1} x_i} > \frac{n}{n-1},$$
where $x_k > 0$, $x_2 > 4x_1$, and $n \ge 2$
- The Weierstraß inequalities (3.2.37 in [16]):
$$\frac{1}{1 - \sum_{k=1}^{n} a_k} > \prod_{k=1}^{n}(1+a_k) > 1 + \sum_{k=1}^{n} a_k,$$
$$\frac{1}{1 + \sum_{k=1}^{n} a_k} > \prod_{k=1}^{n}(1-a_k) > 1 - \sum_{k=1}^{n} a_k$$
$(n \ge 1)$ for $0 < a_k < 1$ with $\sum_{k=1}^{n} a_k < 1$
- An inequality of Beesack [3],
$$\sum_{k=1}^{n}\Big(\sum_{i=1}^{k} x_i\Big)^{\beta} x_i^{\alpha} \le \Big(\sum_{k=1}^{n} x_k\Big)^{\alpha+\beta} \qquad (n \ge 1)$$
for $x_k > 0$ and $\alpha \ge 1, \alpha + \beta \ge 1$, can be done for specific values $\alpha \in \mathbb{N}$, $\beta \in \mathbb{Z}$, e.g., for $\alpha = 2, \beta = -1$.

We are also able to prove inequality 5.16 from [15],

$$\sum_{k=1}^{n}\sin(kx) \ge \frac{1}{2}\sin(nx) \qquad (0 \le x \le \pi, n \ge 1),$$

using

$$\sin((n+2)x) = 2\cos x \sin((n+1)x) - \sin(nx)$$

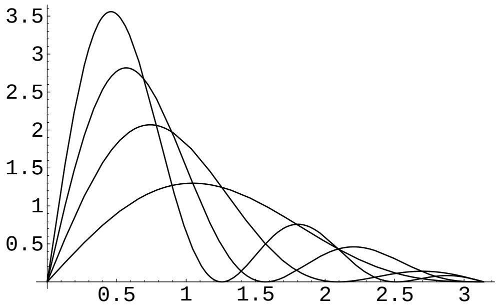
as defining recurrence and the identity

$$(\cos^2 x - 1) - \sin^2(nx) - 2\cos x \sin(nx)\sin((n+1)x) \\ - \sin^2((n+1)x) = 1 \qquad (n \in \mathbb{N})$$

as well as the facts $-1 \leq \cos x \leq 1$ and $0 \leq \sin(nx) \leq 1$ as additional knowledge. The figure below shows the graph of

$$f_n(x) := \sum_{k=1}^{n} \sin(kx) - \frac{1}{2}\sin(nx)$$

on $0 \leq x \leq \pi$ for $n = 2, 3, 4, 5$, suggesting the validity of the inequality for those special values of n.



Inequalities which are not amenable to our proving procedure can sometimes be rewritten in a way that makes the proving procedure applicable. As an example, consider inequality (3) of Knopp and Schur. We have the relation

$$\sum_{k=1}^{\infty} \frac{1}{k^2\binom{k+n}{n}} = \psi'(n+1) = \frac{\pi^2}{6} - \mathrm{H}_n^{(2)}$$

where the first identity was provided by Mathematica, and the second follows from Thm. 1.2.7 in [1]. $\mathrm{H}_n^{(2)} = \sum_{k=1}^{n} 1/k^2$ denotes the harmonic number of second order and ψ the digamma function [1]. Zeilberger's algorithm [20, 21] delivers

$$\sum_{k=1}^{n-1} \frac{1}{k^2\binom{k+n}{n}} = \sum_{k=1}^{n-1} \frac{3k^2+3k+1}{2(k+1)(2k+1)k^2\binom{2k-1}{k-1}} - \frac{1}{(k+1)^2}$$

by which (3) simplifies to

$$\frac{\pi^2}{6} - 1 - \sum_{k=1}^{n-1} \frac{3k^2+3k+1}{k^2(k+1)(2k+1)\binom{2k}{k}} < \frac{2}{n\binom{2n}{n}},$$

and this is indeed in the scope of our method. A minor obstacle is the fact that π is not algebraic and might cause difficulties in the CAD computations, but these are easily circumvented by regarding π as a parameter with the restriction $3 < \pi < 4$. In this setup the proof is successful.

The examples presented in this section are only a small selection of a large number of inequalities, mainly from [15, 16], that we were able to verify by means of our method. A collection of further examples is provided in form of a Mathematica notebook which is available on the world wide web [11]. Also details about the automated proofs for all the inequalities mentioned in this paper can be found in this file.

4.2 Difficult Examples

There are some prominent examples of quite difficult inequalities that also fit well into the class of inequalities we consider. One example is the Askey-Gasper inequality mentioned in the introduction. This inequality reduces for $\alpha = 0$ to Fejér's inequality [1]. Another example is the inequality of Vietoris [15, § 0.8]: if $(a_n)_{n\geq 1}$ is positive, decreasing, and satisfies

$$a_{2k} \leq \frac{2k-1}{2k} a_{2k-1} \quad (1 \leq k \leq n/2),$$

then

$$\sum_{k=1}^{n} a_k \sin(kx) > 0 \quad (0 < x < \pi).$$

Choosing $a_k = 1/k$ gives the Fejér-Jackson inequality [15]

$$\sum_{k=1}^{n} \frac{\sin(kx)}{k} \geq 0 \quad (0 < x < \pi, n \geq 1).$$

As opposed to the examples of Section 4.1, which could all be completed with a negligible amount of CPU time, we did not succeed in proving any of these outstanding inequalities. Already in the second iteration of the proving procedure, the CAD computations exceed our memory capacity (1Gb). We have tried both Mathematica's CAD implementation and the QEPCAD system [8]. It is not clear whether the procedure terminates on any of these examples.

5. APPLICATION: SIGN PATTERNS OF C-FINITE SEQUENCES

The procedure of Section 3 can be modified slightly in order to analyze the sign patterns of oscillating sequences. Consider *C-finite* sequences $(f_n)_{n\geq 0}$ defined by linear homogeneous recurrences with constant coefficients, for example

$$f_0 = 2+\sqrt{2}, \quad f_1 = 2+\sqrt{10}, \quad f_2 = -2+5\sqrt{2} \\ f_n = (4+\sqrt{5})f_{n-1} - (5+4\sqrt{5})f_{n-2} + 5\sqrt{5}f_{n-3} \quad (n > 3).$$

The initial values and recurrence coefficients are chosen such that $(f_n)_{n\geq 1}$ has the closed form

$$f_n = \sqrt{2}\, 5^{n/2}(1 - 2\sin(n\theta - \tfrac{\pi}{4})) \qquad (n \geq 0)$$

with $\theta = \arctan\frac{1}{2}$. It is well known [4] that the numbers $(n\theta - \frac{\pi}{4}) \bmod 2\pi$ lie dense in the interval $[0, 2\pi]$, hence f_n clearly has infinitely many positive and infinitely many negative values. Our goal is to obtain finer information on the sign of f_n.

As additional knowledge, we use the identity

$$25f_n^2 - \tfrac{10}{11}(14+13\sqrt{5})f_nf_{n+1} - \tfrac{20}{11}(2-6\sqrt{5})f_nf_{n+2} \\ + (6+4\sqrt{5})f_{n+1}^2 + f_{n+2}^2 - \tfrac{2}{11}(14-13\sqrt{5})f_{n+1}f_{n+2} = 0,$$

which was found by an ansatz with undetermined coefficients and verified by the algorithm proposed in [13].

In order to study the sign pattern of $(f_n)_{n\geq 1}$, we use the procedure of Section 3 to prove that a certain sequence of sign changes determines the sign of the next value. Indeed, if $C\{t\}$ is a difference field where t represents the sequence $(f_n)_{n\geq 1}$, then

$$\mathrm{Reduce}(\{\mathrm{id}\}, \{t \geq 0, s(t) < 0\}, \{s^2(t) \geq 0\}) = \mathrm{false},$$

where id is the identity above and Mathematica's CylindricalDecomposition is again used for Reduce. Hence, if $f_n \geq 0$ and $f_{n+1} < 0$, then also $f_{n+2} < 0$. Let us write this as $(+|0)- \;\Rightarrow -$. In the same way, we can show that

$$
\begin{aligned}
&(+|0)--\Rightarrow -,\\
&(+|0)---\;\Rightarrow -,\\
&+++++(+|0)----*\;\Rightarrow +,\\
&(+|0)----*+\Rightarrow +,\\
&(+|0)----*++\;\Rightarrow +,\\
&(+|0)----*+++\Rightarrow +,\\
&(+|0)----*++++\;\Rightarrow +,\\
&(+|0)----*+++++\Rightarrow +,\\
&(+|0)----*++++++\;\Rightarrow +,\\
&(+|0)----*+++++++\Rightarrow +,\\
&(+|0)----(-|0)++++++++\Rightarrow +,\\
&(+|0)----*++++++++*\Rightarrow -
\end{aligned}
$$

where $*$ may be $+$, 0, or $-$. Putting all these rules together, we find that the sign pattern of $(f_n)_{n\geq 1}$ may be written in the form

$$+^9[+\mid 0]-^4[-\mid 0],$$

i.e., there are alternating runs of nine positive and four negative values, but each run may be prolonged by one, or there might be a zero between two runs. (Following standard notation, we write $[a]$ for "a or nothing," $a \mid b$ for "a or b," and a^n for n repetitions of a.)

In fact, the first values of the sequence have the signs

$$+^3-^5+^9-^4+^9-^5+^9-^4+^{10}-^4+^9-^5+^9-^4+^9\cdots$$

which is in accordance with the predicted pattern.

What can be said about the optional prolongations $[+\mid 0]$ and $[-\mid 0]$ in the pattern? When do they occur? By a more careful application of the same technique, we obtained the following refined description of the sign pattern:

$$(0\mid +)+^9P^5-^5+^9P^4-^4\big[-+^9P^4\big[P(-^5+^9P^4)^2\big]-^4\big]$$

where P stands for the subpattern $-^4+^9-^5+^9$. This gives much more detail compared to the first pattern, even though it still doesn't describe the pattern entirely. In this pattern, there are only two uncertain signs per 674 values left.

The computations we did for the sign pattern analysis can also be found in the Mathematica notebook [11].

6. CONCLUSION

Though inequalities are considered much harder than identities from a computer algebra viewpoint, inequalities are not entirely out of the scope of symbolic methods. In this paper, we have contributed a simple but powerful method for proving special function inequalities, based on a connection of CAD with difference algebra. The method is strong enough to succeed on a wide range of interesting examples using only little – if any – support by the user. However, termination is not guaranteed in general, and it would be interesting to have a description of the class of inequalities where the procedure terminates. There might be manageable conditions on the input recurrences that ensure termination, and the class might enjoy some useful closure properties.

7. REFERENCES

[1] G. E. Andrews, R. Askey, and R. Roy. *Special Functions*, volume 71 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 1999.

[2] R. Askey and G. Gasper. Positive Jacobi polynomial sums II. *American Journal of Mathematics*, 98:709–737, 1976.

[3] P. R. Beesack. On certain discrete inequalities involving partial sums. *Canadian Journal of Mathematics*, 21:222–234, 1969.

[4] J. W. S. Cassels. *An Introduction to Diophantine Approximation*. Cambridge University Press, 1957.

[5] B. Caviness and J. R. Johnson, editors. *Quantifier Elimination and Cylindrical Algebraic Decomposition*, Texts and Monographs in Symbolic Computation. Springer, 1998.

[6] R. M. Cohn. *Difference Algebra*. Interscience Publishers, John Wiley & Sons, 1965.

[7] G. E. Collins. Quantifier elimination for the elementary theory of real closed fields by cylindrical algebraic decomposition. *Lecture Notes in Computer Science*, 33:134–183, 1975.

[8] G. E. Collins and H. Hong. Partial cylindrical algebraic decomposition for quantifier elimination. *Journal of Symbolic Computation*, 12(3):299–328, 1991.

[9] L. de Branges. A proof of the Bieberbach conjecture. *Acta Mathematica*, 154:137–152, 1985.

[10] S. B. Ekhad. A short, elementary, and easy, WZ proof of the Askey-Gasper inequality that was used by de Branges in his proof of the Bieberbach conjecture. *Theoretical Computer Science*, 117:199–202, 1993.

[11] S. Gerhold and M. Kauers. A procedure for proving special function inequalities involving a discrete parameter – Mathematica notebook with examples. Technical Report 2005-02, SFB F013, Johannes Kepler Universität, 2005. http://www.risc.uni-linz.ac.at/research/combinat/publications/.

[12] G. Hardy, J. E. Littlewood, and G. Pólya. *Inequalities*. Cambridge Mathematical Library. Cambridge University Press, second edition, 1952.

[13] M. Kauers. An algorithm for deciding zero equivalence of nested polynomially recurrent sequences. Technical Report 2003-48, SFB F013, Johannes Kepler Universität, 2003. (submitted).

[14] M. Kauers. Computer proofs for polynomial identities in arbitrary many variables. In *Proceedings of ISSAC '04*, pages 199–204, July 2004.

[15] D. S. Mitrinović. *Elementary Inequalities*. P. Noordhoff Ltd., 1964.

[16] D. S. Mitrinović. *Analytic Inequalities*. Springer, 1970.

[17] T. S. Nanjundiah. Problem 10347. *The American Mathematical Monthly*, 100(10), 1993.

[18] P. Paule. A proof of a conjecture of Knuth. *Experimental Mathematics*, 5:83–89, 1996.

[19] P. Paule. A computerized proof of $\zeta(2)=\pi^2/6$. *In preparation*, 2005.

[20] P. Paule and M. Schorn. A Mathematica version of Zeilberger's algorithm for proving binomial coefficient identities. *Journal of Symbolic Computation*, 20(5–6):673–698, 1995.

[21] M. Petkovšek, H. Wilf, and D. Zeilberger. *$A = B$*. AK Peters, Ltd., 1997.

[22] I. Schur and K. Knopp. Über die Herleitung der Gleichung $\sum_{n=1}^{\infty} 1/n^2 = \pi^2/6$. *Archiv der Mathematik und Physik*, 3(27):174–176, 1918.

[23] J. M. Steele. *The Cauchy-Schwarz Master Class. An Introduction to the Art of Mathematical Inequalities.* Cambridge University Press, 2004.

[24] A. Strzeboński. Solving systems of strict polynomial inequalities. *Journal of Symbolic Computation*, 29:471–480, 2000.