Student seminar on Automatic proofs of binomial identities (UZH-FS2020)

Algorithm Hyper (Part II)

Presentation of Chapters 8.5. to 8.10 in
***A=B* , by M. Petkovsek, H. Wilf, D. Zeilberger**

Zouhair Ouaggag, UZH, 28.5.2020

Content:

1. Implementations of Algorithm Hyper in Mathematica:
    a) Outline of Algorithm Hyper
    b) Finding hypergeometric solutions:
        - Example 1: all solutions in $\mathcal{H}(F)$
        - Example 2: all solutions in $\mathcal{H}(K)$
2. Finding all hypergeometric solutions and more
    a) Reducing the order of recurrence, factorization of linear recurrence operators
    b) Solutions in d'Alembertian sequences
        - Example 3: all solutions in $\mathcal{L}(H)$
    c) Finding all solutions in $\mathcal{L}(H)$

1. Implementations of Algorithm Hyper in Mathematica

a) Outline of Algorithm Hyper

- Goal: solve $Ly = 0$

  where $L = \sum_{i=0}^{r} p_i(n) N^i$, $p_i \in \mathcal{P}(F)$, $y \in \mathcal{H}(K)$
  K a field extension of F

- Algorithm: (for second-order linear recurrence)

  consider $p_2(n)y(n+2) + p_1(n)y(n+1) + p_0(n)y(n) = 0$

1. Implementations of Algorithm Hyper in Mathematica

- Algorithm: (for second order linear recurence)

consider $$p_2(n)y(n+2)+p_1(n)y(n+1)+p_0(n)y(n)=0$$

1. since solution $y \in \mathcal{H}(K)$, we set $y(n+1)=S(n)y(n)$ for some $S \in \mathcal{R}(K)$, subsitute in the equation and cancel $y(n)$:

$$p_2(n)S(n+1)S(n)y(n)+p_1(n)S(n)y(n)+p_0(n)y(n)=0$$

$$p_2(n)S(n+1)S(n)+p_1(n)S(n)+p_0(n)=0$$

2. write $S(n)$ in the canonical form (Thm 5.3.1.) and substitute in the equation:

$$S(n)=z\frac{a(n)}{b(n)}\frac{c(n+1)}{c(n)}$$

$$z^2p_2(n)a(n+1)a(n)c(n+2)+zp_1(n)b(n+1)a(n)c(n+1)+p_0(n)b(n+1)b(n)c(n)=0$$

1. Implementations of Algorithm Hyper in Mathematica

- Algorithm: (for second order linear recurence)

$$z^2 p_2(n)a(n+1)a(n)c(n+2) + zp_1(n)b(n+1)a(n)c(n+1) + p_0(n)b(n+1)b(n)c(n) = 0$$

3. $a(n)$ is relatively prime with $c(n)$, $b(n)$and $b(n+1)$, hence $a(n) \mid p_0(n)$. Similarly, $b(n+1) \mid p_2(n)$. So we cancel factors $a(n)$ and $b(n+1)$ in the p_i's :

$$z^2 \frac{p_2(n)}{b(n+1)} a(n+1)c(n+2) + zp_1(n)c(n+1) + \frac{p_0(n)}{a(n)} b(n)c(n) = 0$$

4. For each choice of $a(n)$ and $b(n)$, equate the leading coefficient in LHS to zero and solve quadratic equation for z .

5. For each choice of $a(n)$, $b(n)$ and z, solve auxiliary recurrence equation for $c(n)$, using Algorithm Poly.

1. Implementations of Algorithm Hyper in Mathematica

Remarks:

- generalization to higher order recurrences: Algorithm Poly in step 6 (method of undetermined coefficients) to a larger SLE.

- Maximal complexity if no hypergeometric solution exits (need to check all possible triples $(a(n), b(n), z)$.

- Check triples $(a(n), b(n), z)$ until generating $Ker(L) \cap \mathcal{L}(\mathcal{H}_K)$ (Note: $Ker(L)$ may need more than just hypergeometric sequences to be spanned).

- Coefficient $p_i's$ are in $\mathcal{P}(F)$, but hypergeometric solutions $\mathrm{y}(n)$ are in $\mathcal{H}(K)$, because:
    - Splitting field for the $p_i's$
    - Spliting fields of the polynomial to solve for the constant z.

- Algorithm Hyper in Mathematica stops after first solution.
  → all solutions in $\mathcal{H}(K)$ by trying all triples $(a(n), b(n), z)$.
  → yields a generating set of $Ker(L) \cap \mathcal{L}(\mathcal{H}_K)$, not necessarily a basis.

1. Implementations of Algorithm Hyper in Mathematica

a) Finding all hypergeometric solutions:

Example 1: $$y(n+2) - 2(n+2)y(n+1) + (n+1)(n+2)y(n) = 0$$

- Algorithm Hyper gives a first solution:

  -In[1]:= Hyper[y[n + 2] - 2 (n + 2) y[n + 1] + (n + 1) (n + 2) y[n] == 0, y[n]]

  -Out[1]= {2 + n}

  i.e. $S(n) = \frac{y(n+1)}{y(n)} = n + 2$ and a hypergeometric solution is $y(n) = (n+1)!$.

- Calling all hypergeometric solutions gives:

  -In[2]:= Hyper[y[n + 2] - 2 (n + 2) y[n + 1] + (n + 1) (n + 2) y[n] == 0, y[n], Solutions → All]

  -Out[2]= {1 + n, (1 + n)^2/n, 2 + n}

- This gives the generating set:

  $\{ n!\, ; \frac{(n!)^2}{(n-1)!}\, ; (n+1)! \}$ containing the basis $\{ n!\, ; \frac{(n!)^2}{(n-1)!} \}$. (the 3rd solution is the sum of the first two).

1. Implementations of Algorithm Hyper in Mathematica

a) <u>Finding all hypergeometric solutions:</u>

Example 2: $$y(n+2) - (2n+1)y(n+1) + (n^2-3)y(n) = 0$$

- Algorithm Hyper gives a first solution:

  -In[1]:= Hyper[y[n + 2] - (2 n + 1) y[n + 1] + (n^2 - 3) y[n] == 0, y[n]
  Warning: irreducible factors of degree > 1 in trailing coefficient; some solutions may not be found
  -Out[1]= {}

  i.e. no solution in $\mathcal{H}(\mathbb{Q})$

  → consider splitting field of trailing coefficient p_0, $\mathbb{Q}(\sqrt{3})$

- Calling all hypergeometric solutions in $\mathcal{H}(\mathbb{Q}(\sqrt{3})$ gives:

  -In[2]:= Hyper[y[n + 2] - 2 (n + 2) y[n + 1] + (n + 1) (n + 2) y[n] == 0, y[n], Quadratics → True, Solutions → All]
  -Out[2]= {- $\sqrt{3}$ + n, $\sqrt{3}$ + n}

  i.e. $S(n) = \frac{y(n+1)}{y(n)} = \pm\sqrt{3} + n$ and hypergeometric solutions is $y(n) = \left(\pm\sqrt{3}\right)_n$.

2. Finding all hypergeometric solutions and more

- Considering all possible triples $(a(n), b(n), z)$ in Algo Hyper

    → all solutions in $\mathcal{H}(K)$

    → actually a generating set for $Ker(L) \cap \mathcal{L}(\mathcal{H}_K)$

- Another method:
    - find a first hypergeometric solution
    - reduce the order of the recurrence equation (i.e. factorize the recurrence operator)
    - look for a hypergeometric solution for the reduced recurrence,
    - … (repeat)

    → factorization of the linear recurrence operator L as a composition of first-order linear recurrence operators $L = L_k L_{k-1} \dots L_1$.

    → solutions from a larger class of sequences, the *d'Alembertian sequences*,

    i.e. sequence of the form $y = h_1 \sum(h_2 \sum(\dots \sum h_k))$,

    where $y = \sum h$ means $\Delta y(n) = y(n+1) - y(n) = h(n)$

2. Finding all hypergeometric solutions and more

a) Outline of reducing order of L

consider a second-order linear recurrence:

$$p_2(n)y(n+2)+p_1(n)y(n+1)+p_0(n)y(n)=0$$

1. Algo Hyper gives a first hypergeometric solution $h_1 \in \mathcal{H}(K)$.
   Note: $L_1 h_1 = 0$, for some L_1 of first order (since by definition $p_1(n)h_1(n+1)+p_0(n)h_1(n)=0$, for some p_0 and $p_1 \in \mathcal{P}(K)$.

2. look for a solution of the form $y(n)=h_1 \sum h_2$, where $h_2 \in \mathcal{H}(K)$.
   → substituting for $y(n)$ in $Ly=0$
   → a first order recurrence equation which admits $h_2 \in \mathcal{H}(K)$ as a solution.

3. We obtained two solutions h_1 and $h_1 \sum h_2$ which span $Ker(L) \cap \mathcal{A}(\mathcal{H}_K)$.
   (Note that $h_1 \sum h_2$ is not necessarily hypergeometric.)

2. Finding all hypergeometric solutions and more

Remarks:

- Euclidean division in the ring of linear recurrence operators with coefficients in $\mathcal{R}(K)$:
  - shift operator N distributes and commutes with linear recurrence operators (rule: $Np(n) = p(n+1)N$),
    → define a multiplication and a division with rest of linear recurrence operators.
  - Let $y \in \mathcal{S}(K)$ s.t. $Ly = 0$, and let M of minimal order such that $My = 0$.
    → division of L by M: $L = QM + R$, Q and R linear recurrence operators, ord(R) < $\mathrm{ord}(M)$.
    Since L and M annihilate y, so does R which is then necessarily the zero operator.
    Hence the operator with minimal order annihilating y is a factor of the operator L.
    Conversely, if $My = 0$ then obviously $QMy = 0$ for any Q.
- solving $Ly = 0$ is equivalent to factorizing the linear operator as $L = L_2L_1$ where L_1 is the annihilating operator of y of minimal order.
  When y is hypergeometric the minimal operator L_1 is of first order.

2. Finding all hypergeometric solutions and more

b) Solutions in d'Alembertian sequences:

Example: $y(n) = \sum_{k=0}^{n} \binom{3k}{k} \binom{3n-3k}{n-k}$

- Zeilberger Algo gives L such that $Ly = 0$:

$$L = 8(n+2)(2n+3)N^2 - 6(36n^2 + 99n + 70)N + 81(3n+2)(3n+4)$$

- Algo Hyper gives a unique solution in $Ker(L) \cap \mathcal{H}(K)$:
  - -In[2]:= Hyper[8 (n + 2) (2 n + 3) y[n + 2] - 6 (36 n^2 + 99 n + 70) y[n + 1] + 81 (3 n + 2) (3 n + 4) y[n] == 0, y[n], Solutions → All]
  - -Out[2]= {27/4}

  i.e. $S(n) = \frac{y(n+1)}{y(n)} = \frac{27}{4}$ and the fundamental hypergeometric solution is $y(n) = \left(\frac{27}{4}\right)^n$.

- We divide L by the factor $L_1 = 4N - 27$ corresponding to the solution above,

  $\rightarrow L = L_2 L_1$, with $L_2 = 2(n+2)(2n+3)N - 3(3n+2)(3n+4)$.

2. Finding all hypergeometric solutions and more

b) Solutions in d'Alembertian sequences:

- We divide L by the factor $L_1 = 4N - 27$ corresponding to the solution above,

  $\rightarrow \quad L = L_2 L_1$, with $L_2 = 2(n+2)(2n+3)N - 3(3n+2)(3n+4)$.

- The first order recurrence $L_2 y = 0$ writes $y(n+1) = \frac{3(3n+2)(3n+4)}{2(n+2)(2n+3)} y(n)$

  $\rightarrow$ hypergeometric solution $y(n) = \frac{\binom{3n}{n}}{3n-1}$.

- Hence a second fundamental solution for the first recurrence equation is:

$$y(n) = \left(\frac{27}{4}\right)^n \sum_{k=0}^{n} \frac{\binom{3k}{k}}{3k-1} \left(\frac{27}{4}\right)^{-k}$$

- Considering the initial values $y(0) = 1$ and $y(1) = 6$ we get the solution:

$$y(n) = \frac{1}{2}\left(\frac{27}{4}\right)^n \left(1 - \sum_{k=0}^{n} \frac{\binom{3k+3}{k+1}}{3k+2} \left(\frac{27}{4}\right)^{-k}\right)$$

2. Finding all hypergeometric solutions and more

b) <u>Finding all solutions in $\mathcal{L}(\mathcal{H}_K)$:</u> (i.e. all closed form solutions)

Note:

- If $h \in \mathcal{H}_K$ $(Lh \neq 0)$, then $Lh \in \mathcal{H}_K$ and is similar to h

  (write $r = Nh/h$, substitute in Lh and factor out h)

- If $h \in \mathcal{H}_K$ s.t. $Lh = 0$ and $h = \sum_{i=1}^{k} h_i$ for h_i's pairwise dissimilar, then $Lh_i = 0$ for all i

  (if $Lh_i \neq 0$ write $Lh_i = r_i h_i$, substitute in $Lh = 0$, yields $\sum_{i=1}^{k} h_i = 0$ then $r_i = 0$ since h_i are dissimilar)

- If $h \in \mathcal{L}(\mathcal{H}_K)$ s.t. $Lh = 0$ then $h = \sum_{i=1}^{k} h_i$ for h_i's s.t. $Lh_i = 0$ for all i.

  sum of similar hypergeometric terms is either hypergeometric or zero,
  so write $h = \sum_{i=1}^{k} h_i$ with h_i's pairwise dissimilar hypergeometric,
  then substitute in $Lh = 0$, then follows $Lh_i = 0$ for all i by previous remark.

It follows that all solutions in $\mathcal{L}(\mathcal{H}_K)$ are spanned by the solutions in $\mathcal{H}_K$ obtained by Algorithm Hyper.

Thank you