Finite Summation of Integer Powers x^p, Part 3

Assad Ebrahim * Carol Ouellette†

2/24/2010
(Rev. 0i, April 4, 2010)

Abstract

We find a *direct* closed-form solution, i.e. one that does *not* require iteration, for the general case of the finite-summation-of-integer-powers problem $S_p(N) = \sum_{k=1}^{N} k^p$. Having established in Part 2 [Ebr10b] that the closed-form solution is a polynomial, the summation is here rewritten as the sum of the $p+1$ independent monomials $a_j N^j$ $(1 \leq j \leq p+1)$, where the a_j are unknown coefficients. Using the recurrence relation $S_p(N+1) = S_p(N)+(N+1)^p$, we obtain a linear combination of the monomials, which reduces to an easily solvable $(p+1)$-by-$(p+1)$ triangular linear system in the unknown coefficients a_j of the closed-form polynomial solution. Maxima and Octave/Matlab codes for directly computing the closed-form solutions are included in the Appendices.

Finding a Closed-Form solution for $S_p(N) = \sum_{k=1}^{N} k^p$ using a Direct Method

Motivation

Our goal is to obtain, without using iteration, a closed form solution for the general case of the finite-summation-of-integer-powers problem:

$$S_p(N) = \sum_{k=1}^{N} k^p, \text{ (where } p \in \mathbb{N} = \{0, 1, 2, \ldots\}; N \geq 0) \tag{1}$$

Recall that in Part 2 [Ebr10b] of this paper, we used a method of recurrence relations to obtain the following results about $S_p(N)$:

*Assad Ebrahim received his M.Sc. in Applied Mathematics from the University of Washington in Seattle and his B.A. in Mathematics from Swarthmore College in Philadelphia. He is Director of Engineering and Operations at BioSonics, Inc., a company specializing in the design and manufacture of sonar systems and software for the detection, analysis, and classification of sonar signals applied to fisheries research, environmental monitoring, and homeland defense. (http://www.biosonicsinc.com). He is also the Founder of Mathematical Science & Technologies (http://www.mathscitech.org), an organization involved in the application of mathematics, technology, and applied science to industry, and in the enhancement of curricula for STEM education (Science, Technology, Engineering, and Mathematics). Contact Information: assad.ebrahim@mathscitech.org

†Carol Ouellette received her M.Sc. in Electrical Engineering from the Massachusetts Institute of Technology and her B.Sc./B.A. in Engineering and Mathematics from Swarthmore College in Philadelphia. She has worked as an R&D Engineer on sensor systems, sensor fusion, and missile tracking systems at several defense companies, including AlphaTech, Lincoln Labs, and Raytheon. She is a guest writer and consulting scientist at MathSciTech. Contact Information: carol.ouellette@mathscitech.org

1. $S_p(N)$ may be expressed as a linear recurrence relation that uses the $p-1$ lower order formulas $S_j(N)$ and $S_j(N-1)$ $(j = 1, \ldots, p-1)$ as follows:

$$S_p(N) = \frac{1}{1+\alpha_p}\left\{N^2 + N^p + \sum_{j=1}^{p-1} C_j(N)S_j(N-1) - \sum_{j=1}^{p-1}\alpha_j S_j(N)\right\}. \tag{2}$$

   The α_j in (2) are the coefficients from the $(p-1)$-st polynomial solution $S_{p-1}(N)$, while the terms $C_j(N)$ are defined in terms of binomial coefficients[1] as:

$$C_j(N) = \left[\binom{p-1}{j}N - \binom{p-1}{j-1}\right].$$

2. The closed form solution of $S_p(N)$ is a polynomial in N of degree $p+1$ with rational coefficients and a constant term equal to zero.

While the expression for $S_p(N)$ in (2) is indeed accurate, it requires repeated iteration to obtain the closed form expression for any particular p. Such an approach without the assistance of a computer algebra system such as Maxima[2] would be prohibitively time-consuming and prone to error.

We are hence left with the following question:

> Can we find a closed-form solution for the coefficients of the general solution polynomial that can be obtained *directly*, i.e. that does not require iteration?

This paper uses linear algebra and matrices to achieve precisely this goal.

The Direct Approach

Polynomial with Undetermined Coefficients

Since we have established in (2) that $S_p(N)$ is a polynomial of order $p+1$ with no constant term, we may write down the closed form solution of $S_p(N)$ in the form:

$$S_p(N) := \sum_{j=1}^{p+1} a_j N^j \tag{3}$$

where a_j, $(j = 1,\ 2,\ \ldots,\ p+1)$, are coefficients that have yet to be determined. As a first step towards determining these coefficients, observe that every finite summation can be written as a first order recurrence in N by peeling off the last summand:

$$S_p(N+1) = S_p(N) + (N+1)^p. \tag{4}$$

Substituting (3) into (4) gives:

$$\sum_{j=1}^{p+1} a_j(N+1)^j = \sum_{j=1}^{p+1} a_j N^j + (N+1)^p. \tag{5}$$

[1] The notation $\binom{n}{k}$ used above denotes binomial coefficients, often expressed verbally as "n choose k". Other representations of these coefficients include $C(n,k)$ and C^n_k.

[2] Maxima code for automatically computing solutions to (1) using solution formula (2) is given in Part 2 of this paper [Ebr10b].

Summation Manipulations

What follows is a sequence of summation manipulations[3] and simplifications of (5) aimed at obtaining an expression from which a closed-form solution for the coefficicents a_j becomes transparent:

I Expand the binomial powers using the binomial formula on both the left-hand side (LHS) and right-hand side (RHS) of (5):

$$\sum_{j=1}^{p+1} a_j \sum_{i=0}^{j} \binom{j}{i} N^i = \sum_{j=1}^{p+1} a_j N^j + \sum_{j=0}^{p} \binom{p}{j} N^j. \tag{6}$$

II Modify the LHS and RHS of (6) as follows, exploiting the fact that $\binom{n}{k} = 0$ when $k > n$:

LHS: Extend upper range of inner sum to match upper range of outer sum.

RHS: Peel off 0-index term and add a vacuous $p+1$-index term to last sum.

Result:

$$\sum_{j=1}^{p+1} a_j \sum_{i=0}^{p+1} \binom{j}{i} N^i = \sum_{j=1}^{p+1} a_j N^j + 1 + \sum_{j=1}^{p+1} \binom{p}{j} N^j$$

III **LHS:** Interchange order of summations.

RHS: Combine terms.

Result:

$$\sum_{i=0}^{p+1} N^i \sum_{j=1}^{p+1} a_j \binom{j}{i} = 1 + \sum_{j=1}^{p+1} \left[a_j + \binom{p}{j} N^j \right]$$

IV **LHS:** Peel off 0-index term from the (now outside) summation on i.

RHS: Change dummy index variable.

Result:

$$\sum_{j=1}^{p+1} a_j + \sum_{i=1}^{p+1} N^i \sum_{j=1}^{p+1} a_j \binom{j}{i} = 1 + \sum_{i=1}^{p+1} \left[a_i + \binom{p}{i} N^i \right]$$

V Bring everything over to the left-hand side to obtain a homogeneous linear combination of the monomials $\{N^i\}_{i=0}^{p+1}$:

$$-1 + \sum_{j=1}^{p+1} a_j + \sum_{i=1}^{p+1} \left[\left(\sum_{j=1}^{p+1} a_j \binom{j}{i} \right) - a_i - \binom{p}{i} \right] N^i = 0 \tag{7}$$

[3]Such manipulations are taught in material such as [GKP] and [Knu].

Linear Independence Reveals Equations for Unknown Coefficients

Since the monomials $\{N^i\}_{i=0}^{p+1}$ are a basis for the vector space $\mathbb{P}_{N+1}$ of polynomials of degree less than or equal to $N+1$, *any linear combination of the* $\{N^i\}_{i=0}^{p+1}$ *will equal zero if and only if each coefficient equals zero.*

Therefore, setting the coefficients from (7) equal to zero gives us a system of $p+2$ simultaneous linear equations in the $p+1$ unknown coefficients a_j, one equation for each of the coefficients of N^i:

$$(i=0) \qquad \sum_{j=1}^{p+1} a_j = 1 \tag{8}$$

$$(i=1,\ldots,p) \qquad \sum_{j=1,j\neq i}^{p+1} \binom{j}{i} a_j = \binom{p}{i} \tag{9}$$

$$(i=p+1) \qquad \sum_{j=1,j\neq p+1}^{p+1} \binom{j}{i} a_j = \binom{p}{i} \tag{*}$$

Observe that (*) reduces to $0=0$, since $j<p+1$ for all j, and $\binom{n}{k}=0$ when $n<k$. Thus we are left with a square system of $p+1$ equations $(i=0,\ldots,p)$ in $p+1$ unknowns a_j, $(j=1,\ldots,p+1)$.

By observing that $\binom{j}{i}=1$ for $i=0$, we can combine (8) and (9) into the single set of linear equations:

$$(i=0,\ldots,p) \qquad \sum_{j=1,j\neq i}^{p+1} \binom{j}{i} a_j = \binom{p}{i} \tag{10}$$

Taking into account the inequality $j\neq i$, and noting once again that $\binom{n}{k}=0$ when $n<k$, (10) simplifies to a triangular linear system:

$$(i=0,\ldots,p) \qquad \sum_{j=i+1}^{p+1} \binom{j}{i} a_j = \binom{p}{i} \tag{11}$$

and hence all $p+1$ equations in the system are linearly independent.

From Linear System to Matrix Equation

The triangular linear system (11) may be readily solved by combining the $p+1$ summations into a single matrix equation and using any one of a number of matrix solver packages. But this requires that the system (11) be re-indexed[4] so that the start value of index i matches that of index j. Following the usual matrix convention, referred to as 1-indexing, we choose the start indices to be $i=j=1$.

The 1-indexed system is:

$$(i=1,\ldots,p+1) \qquad \sum_{j=i}^{p+1} \binom{j}{i-1} a_j = \binom{p}{i-1} \tag{12}$$

[4]Index manipulation is covered in [GKP] and [Knu].

Maxima code for solving the 0-indexed[5] triangular linear system (13) is given in Appendix B. Octave/Matlab code for solving the 1-indexed triangular linear system (12) is given in Appendix C.

The matrix equation equivalent to (12) is: $\mathbf{M} \cdot \mathbf{a} = \mathbf{b}$, where

$$\mathbf{M} := \begin{bmatrix} \binom{1}{0} & \binom{2}{0} & \cdots & \binom{p+1}{0} \\ & \binom{2}{1} & \cdots & \binom{p+1}{1} \\ & \cdots & \binom{j}{i-1} \cdots & \\ & & & \binom{p+1}{p} \end{bmatrix}, \quad \mathbf{a} := \begin{bmatrix} a_1 \\ a_2 \\ \cdots \\ a_{p+1} \end{bmatrix}, \quad \text{and} \quad \mathbf{b} := \begin{bmatrix} \binom{p}{0} \\ \binom{p}{1} \\ \cdots \\ \binom{p}{p} \end{bmatrix}. \quad (14)$$

Since $\mathbf{M}$ is an upper triangular matrix with all non-zero upper triangular entries, we know we can readily solve this for any given values of p and N using a matrix solver system such as Maxima/Mathematica, Octave/Matlab, or Maple, among others. Source code for solutions in Maxima and Octave/Matlab are given in Appendix B and Appendix C, respectively.

Computing the General, Closed-Form Solution

We know from (2) that the general, closed-form solution to (1) is a $p+1$-degree polynomial (3) in N with rational coefficients a_j given by solving the triangular linear system (12), or equivalently (14).

The highest few coefficients a_j can be readily computed by hand. For all p, we have:

1. $a_{p+1} = \frac{1}{p+1}$
2. $a_p = \frac{1}{2}\binom{p}{0} = \frac{1}{2}$
3. $a_{p-1} = \frac{1}{12}\binom{p}{1} = \frac{p}{12}$
4. $a_{p-3} = -\frac{1}{120}\binom{p}{3}$
5. $a_{p-5} = \frac{1}{252}\binom{p}{5}$
6. $a_{p-7} = -\frac{1}{240}\binom{p}{7}$
7. $a_{p-9} = \frac{1}{132}\binom{p}{9}$.
8. In particular, we claim (without proof) that

$$a_{p-i} = c_i \binom{p}{i}, \quad (15)$$

where the c_i are rational coefficients independent of p, and $c_i = 0$ for all even positive i. Hence we have:

$$a_{p-2} = a_{p-4} = \ldots = 0.$$

[5] 0-indexing is more convenient for various computational software packages and programming langauges, including Maxima. The 0-indexed system is:

$$(i = 0, \ldots, p) \qquad \sum_{j=i}^{p} \binom{j+1}{i} a_j = \binom{p}{i} \quad (13)$$

Substituting the above coefficients into the general closed-form polynomial solution gives us:

$$S_p(N) = \frac{1}{p+1}N^{p+1} + \frac{1}{2}N^p + \frac{p}{12}N^{p-1} - \frac{1}{120}\binom{p}{3}N^{p-3} + \frac{1}{252}\binom{p}{5}N^{p-5} - \frac{1}{240}\binom{p}{7}N^{p-7} + \frac{1}{132}\binom{p}{9}N^{p-9} - \ldots + pc_{p-1}N. \quad (16)$$

A listing of the solution formulas for the first ten p values is given in Appendix A.

Conclusion

Reviewing the path we have taken in this 3-part paper: Part 1 [Ebr10a] motivated the problem and illustrated a recurrence relation method for obtaining the solution for small p. Part 2 [Ebr10b] generalized this method and found a p-th order recurrence relation for the solution. Part 2 also illustrated, by an induction argument, that the closed form solutions for all p are polynomials of degree $p + 1$ with rational coefficients and no constant term.

This paper used the closed-form polynomial expression motivated in Part 2 to obtain a direct solution. By writing out the polynomial form with undetermined coefficients, the recurrence relation (4) was manipulated into a linear combination of polynomials (7). Using a linear independence argument, (7) was reduced to a triangular linear system (12) and associated matrix equation (14). The general closed-form solution is (16).

Solving the finite-summation-of-integer-powers problem $\sum_{k=1}^{N} k^p$, for arbitrary positive integers p and N has provided a natural setting to use a variety of techniques from discrete mathematics and linear algebra, and poses additional interesting questions (characterization of the denominators in the coefficients of (16) (the c_i in (15)) and divisibility properties of S_p for various p). In particular, we have obtained a direct method for solving the finite-summation-of-integer-powers problem that invokes a matrix solution and does not require iteration.

Appendix A: Listing of Solutions to (1)

A listing of the solution formulas for the first ten p values is as follows:

$$S_1(N) = \sum_{k=1}^{N} k = \frac{N^2}{2} + \frac{N}{2}$$

$$S_2(N) = \sum_{k=1}^{N} k^2 = \frac{N^3}{3} + \frac{N^2}{2} + \frac{N}{6}$$

$$S_3(N) = \sum_{k=1}^{N} k^3 = \frac{N^4}{4} + \frac{N^3}{2} + \frac{N^2}{4}$$

$$S_4(N) = \sum_{k=1}^{N} k^4 = \frac{N^5}{5} + \frac{N^4}{2} + \frac{N^3}{3} - \frac{N}{30}$$

$$S_5(N) = \sum_{k=1}^{N} k^5 = \frac{N^6}{6} + \frac{N^5}{2} + \frac{5\,N^4}{12} - \frac{N^2}{12}$$

$$S_6(N) = \sum_{k=1}^{N} k^6 = \frac{N^7}{7} + \frac{N^6}{2} + \frac{N^5}{2} - \frac{N^3}{6} + \frac{N}{42}$$

$$S_7(N) = \sum_{k=1}^{N} k^7 = \frac{N^8}{8} + \frac{N^7}{2} + \frac{7\,N^6}{12} - \frac{7\,N^4}{24} + \frac{N^2}{12}$$

$$S_8(N) = \sum_{k=1}^{N} k^8 = \frac{N^9}{9} + \frac{N^8}{2} + \frac{2\,N^7}{3} - \frac{7\,N^5}{15} + \frac{2\,N^3}{9} - \frac{N}{30}$$

$$S_9(N) = \sum_{k=1}^{N} k^9 = \frac{N^{10}}{10} + \frac{N^9}{2} + \frac{3\,N^8}{4} - \frac{7\,N^6}{10} + \frac{N^4}{2} - \frac{3\,N^2}{20}$$

$$S_{10}(N) = \sum_{k=1}^{N} k^{10} = \frac{N^{11}}{11} + \frac{N^{10}}{2} + \frac{5\,N^9}{6} - N^7 + N^5 - \frac{N^3}{2} + \frac{5\,N}{66}$$

Appendix B: Maxima Source Code for Solving (14)

For particular p, the coefficients a_j for the closed form solutions $S_p(N)$ are immediate. Maxima code, is given below, for determining this solution using the 0-indexed triangular linear system (13):

```
solution(p):= block([a, eq],       /* give subroutine variables local scope */
    v : makelist(a[i], i, 0, p),   /* create list of unknowns (0-indexed) */
                                   /* create list of equations (0-indexed) */
   eq : makelist(sum(binom(j+1,i)*a[j],j,i,p) = binom(p,i), i, 0, p),
   linsolve(eq, v)
)$
```

A more elaborate function that computes the series form and factored forms of the solution is given below. This was used to generate the first ten solution listings.

```
 SpN_mat(p):= block([a, eq],     /* give subroutine variables local scope */
    v : makelist(a[i], i, 0, p), /* create list of unknowns (0-indexed)   */
                                 /* create list of equations (0-indexed)  */
```

```
    eq : makelist(sum(binom(j+1,i)*a[j],j,i,p) = binom(p,i), i, 0, p),

 /* find coefficients of solution polynomial by solving linear system */
    sol : linsolve(eq, v),

 /* create polynomial: inner product of {N^i} with coefficients */
    pol : makelist(N^(i+1),i,0,p), /* monomials {N^i} */

 /* closed form formula */
    cff : rhs(sol.pol),  /* inner product of {N^i} with coefficients */
    cff  /* return closed form formula in series */
  )$
```

Appendix C: Octave/Matlab Source Code for Solving (14)

For particular p, the coefficients a_j for the closed form solutions $S_p(N)$ are immediate. Octave/Matlab code, is given below, for determining this solution using the 1-indexed triangular linear system (12):

```
% Function listing coefficients from a_p+1 to a_1 as fractions

function c = sumkp_matrix(p)
    M=zeros(p+1);  % initialize matrix M
    for i=1:p+1
        for j=i:p+1 % set upper triangular elements
            M(i,j)=nchoosek(j,i-1);  %  Equation (12)
        end;
    end;

    b=zeros(p+1,1); % initialize column matrix b
    for i=1:p+1
        b(i)=nchoosek(p,i-1);  %  Equation (12)
    end

    c=inv(M)*b;  % Solve to obtain coefficients

    c=fliplr(c'); % List from highest index to lowest
    sml=abs(c)<1e-8; % Numerical correction: set tiny coeffients to exactly zero
    c(find(sml==1))=0;

    disp("Solution Coefficients (High to Low order):")
    c=rats(c);   % return coefficients as fractions

end;
```

References

[Ebr10a] Assad Ebrahim. Finite summation of integer powers x^p, part 1. January 2010.

[Ebr10b] Assad Ebrahim. Finite summation of integer powers x^p, part 2. February 2010.

[GKP] Ron Graham, Donald Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science.* Addison Wesley.

[Knu] Donald Knuth. *The Art of Computer Programming (3 Volumes).*