KANADE AND RUSSELL’S IDENTITY FINDER

Deniz Küçükahmetler dkucukahmetler@sabanciuniv.edu
Faculty of Engineering and Natural Sciences, Undeclared

Efe Öztaban efeoztaban@sabanciuniv.edu
Faculty of Management, Undeclared

Kağan Kurşungöz kursungoz@sabanciuniv.edu
Faculty of Engineering and Natural Sciences, Mathematics

Abstract

The main aim of this project is understanding integer partitions and various identities of partitions. Also learning generating functions and coding in Maxima is required for this project. Partition identities will be constructed by Maxima code and transformed to a set of number with generating functions. These set of numbers will make easy to realize the patterns in partition identities. With the patterns found, new identities can be generated.

Keywords: Integer Partitions, Partition Identities, Generating Functions.

1 Introduction

Our purpose in this project was coding the algorithms related to integer partitions especially read in the paper by S. Kanade and M. C. Russell titled "Identity Finder and Some New Identities of Rogers–Ramanujan Type" published in Experimental Mathematics, Vol 24, issue 4. Also, our purpose was to realize patterns of integer identities and generating new systems with our constituted background knowledge.

Firstly, we started reading the book, “Integer Partitions” by George Andrews and Kimmo Eriksson for a couple of weeks. While understanding the concept, we questioned the theorems. By the contribution of our weekly meetings, we enlarged our knowledge about integer partitions and integer identities.

Then in the following weeks, we started to learn Maxima for coding the algorithms. We carried our search, simultaneously we developed our skills for Maxima. For instance, creating loops, if-else statements, lists...

In the final weeks, we read the paper (Kanade and Russell), coded theorems found from it and add other theorems that our supervisor suggested. We input numbers to our programme and interpreted the outputs while finding patterns between integers. Our algorithms written in Maxima confirmed the identity equivalences we found in our researches.

2.KANADE AND RUSSELL'S IDENTITY FINDER

2.1 Integer Partitions and Partitions Identities

We started with understanding integer partitions concept. For this purpose, we started to do integer partitions manually and also find number of integer partitions of a given number n. Then we called this function p(n) partition function as in the Andrew's book. Here are the integer partitions and number of partitions of numbers from 1 to 5:

n	Partitions of n	p(n)
1	1	1
2	2 , 1+1	2
3	3 , 2+1 , 1+1+1	3
4	4 , 3+1 , 2+2 , 2+1+1 , 1+1+1+1	5
5	5 , 4+1 , 3+2 , 3+1+1 , 2+2+1 , 2+1+1+1 , 1+1+1+1+1	7

Then we started to work on partitions identities. Partition identities are statements to show "every number has as many integer partitions of this sort as of that sort". They basically are equalities of partition numbers (p(n)) for the same number but with different conditions. And the first identity which we studied was Euler's identity:

$$p(n \mid \text{odd parts}) = p(n \mid \text{distinct parts})$$

Here is the Euler's identity for numbers from 1 to 5:

n	Odd partitions of n	Partitions of n into 1-distinct parts	p(n | odd parts)	p(n | distinct parts)
1	1	1	1	1
2	1+1	2	1	1
3	3 , 1+1+1	3 , 2+1	2	2
4	3+1, 1+1+1	4 , 3+1	2	2
5	5 , 3+1+1 , 1+1+1+1+1	5 , 4+1 , 3+2	3	3

Then we studied Roger-Ramanujan Identity and we tried to construct the N set ourselves. By looking at this N set, we find the pattern as in The Roger-Ramanujan Identity.
Roger-Ramanujan Identity:

$$p(n \mid \text{parts in N}) = p(n \mid \text{2-distinct parts})$$

We started with constructing 2-distinct part table then we tried to construct the N set for 2-distinct parts partition.

n	partitions of n into 2-distinct parts	p(n)
1	1	1
2	2	1
3	3	1
4	4 , 3+1	2
5	5 , 4+1	2
6	6 , 5+1 , 4+2	3
7	7, 6+1 , 5+2	3
8	8 , 7+1 , 6+2 , 5+3	4
9	9 , 8+1 , 7+2 , 6+3 , 5+3+1	5

10	10 , 9+1 , 8+2 , 7+3 , 6+4 , 6+3+1	6

We started by setting N= Ø.

1. There is 1 partition for n=1. There is no partition with N= Ø so 1 must be added to N.
2. There is 1 partition for n=2. There is one partition with N= {1} (1+1) so 2 must not be added to N.
3. There is 1 partition for n=3. There is one partition with N= {1} (1+1+1) so 3 must not be added to N.
4. There is 2 partition for n=4. There is two partition with N= {1} (1+1+1+1) so 4 must be added to N.
5. There is 2 partition for n=5. There is two partition with N= {1,4} (4+1 and 1+1+1+1+1) so 5 must not be added to N.
6. There is 3 partition for n=6. There is two partition with N= {1,4} (4+1+1 and 1+1+1+1+1+1) so 6 must be added to N.
7. There is 3 partition for n=7. There is three partition with N= {1,4,6} (6+1, 4+1+1+1 and 1+1+1+1+1+1+1) so 7 must not be added to N.
8. There is 4 partition for n=8. There is four partition with N= {1,4,6} (6+1+1, 4+4, 4+1+1 and 1+1+1+1+1+1) so 8 must not be added to N.
9. There is 5 partition for n=9. There is four partition with N= {1,4,6} (6+1+1+1, 4+4+1, 4+1+1+1+1+1 and 1+1+1+1+1+1+1+1+1) so 9 must be added to N.
10. There is 6 partition for n=10. There is six partition with N= {1,4,6,9} (9+1, 6+4, 6+1+1+1, 4+4+1+1, 4+1+1+1+1+1+1 and 1+1+1+1+1+1+1+1+1+1) so 10 must not be added to N.

$$N=\{1,4,6,9\ldots\}$$

We see that all the numbers in N set have remainder 1 or 4 when divided by 5. This set of numbers involves all m numbers:

$$m \equiv 1 \text{ or } 4 \pmod 5$$

$$m= 1,4,6,9,11,14,16,19,21,24\ldots$$

So, we showed the explanation of Rogers-Ramanujan Identity with our calculations.

p(n | parts ≡ 1 or 4 (mod 5)) = p(n | 2-distinct parts)

After we studied Rogers-Ramanujan Identity, we looked at 3 distinct parts with a N set.

n	parts in N={1,5,7,1...}	partitions of n into 3-distinct parts
1	1^1	1
2	1^2	2
3	1^3	3
4	1^4	4
5	1^5	5 , 4+1
6	$5^1, 1^6$	6 , 5+1
7	$7^1, 5^1 1^2, 1^7$	7 , 6+1 , 5+2
8	$7^1 1^1, 5^1 1^3, 1^8$	8 , 7+1 , 6+2
9	$7^1 1^2, 5^1 1^4, 1^8$	9 , 8+1 , 7+2 , 6+3

We see that parts in N part is not equal to 3-distinct part when n=9. So, we studied Alder's conjecture to look at these conditions like 3-distinct parts. Alder's inequality:

p(n | parts ≡ ±1 (mod d+3)) ≤ p(n | d-distinct parts)

After we studied these partition identities, we started to use coding in Maxima, which is a computer algebra system suitable for mathematical operations like partitioning integers. So, by using Maxima, we make our calculations faster. We used We code functions for different identities.

```
integer_partitions(5);
{[1,1,1,1,1],[2,1,1,1],[2,2,1],[3,1,1],[3,2],[4,1],[5]}
```

We started with one distinct parts:

```
onedist(t):=   block( w:[],
    for n:0 thru t do(s:integer_partitions(n),
                    x:[],
                    for ptn in s do(
                        a:true,
                        b:-2,
                        for el in ptn do(
                            if el-b<1 and -1<el-b then a:false,
                            b:el),
                    if a=true then x: append(x,[ptn])),
        w: append(w,[length(x)])

),disp(w));
```

```
→       onedist(20);
        [1,1,1,2,2,3,4,5,6,8,10,12,15,18,22,27,32,38,46,54,64]
%o35)  done
```

Then we code for 2-distinct part:

```
twodist(t):=   block( w:[],
   for n:0 thru t do(s:integer_partitions(n),
                  x:[],
                  for ptn in s do(
                     a:true,
                     b:-5,
                     for el in ptn do(
                        if  el-b<2 and -2<el-b then a:false,
                        b:el),
                  if a=true then x: append(x,[ptn])),
      w: append(w,[length(x)])

),disp(w));
```

```
(%i3)   twodist(20);
        [1,1,1,1,2,2,3,3,4,5,6,7,9,10,12,14,17,19,23,26,31]
(%o3)   done
```

2.2 Generating Functions

After understanding the integer partitions and partition identities, we started to work on generating functions which are the power series for keeping the track of the number sequences as in the book (Andrews and Eriksson).

Thereafter, we implemented the algorithm for the theorem written in the paper (Kanade and Russell) using Maxima.

Here is the the theorem and our implementation:

Proposition 1. (Cf. Theorem 10.3 of [Andrews 86].) Let $f(q)$ be a formal power series such that

$$f(q) = 1 + \sum_{n\geq 1} b_n q^n. \tag{2–1}$$

Then

$$f(q) = \prod_{m\geq 1}(1-q^m)^{-a_m}, \tag{2–2}$$

where the a_ms are defined recursively by

$$nb_n = na_n + \sum_{d|n,\, d<n} da_d + \sum_{j=1}^{n-1}\left(\sum_{d|j} da_d\right) b_{n-j}. \tag{2–3}$$

```
f(p):= block (
  a:[],
  for n:1 thru length(p) do (
    if n=1 then a: append(a,[p[n]])
    else(

      sum :0,
      sum : sum + n·p[n],
      for d: 1 thru n−1 do(
        if mod(n, d) = 0 then sum: sum − d·a[d]),
      for j:1 thru n−1 do(
        sum2 : 0,
        for k:1 thru j do(
          if mod(j,k)=0 then sum2: sum2+k·a[k]),
        sum: sum − sum2·p[n−j]),

    sum : sum/n,
    a : append(a, [sum]))),
  disp(a));
```

By this implementation we generated the generating function f(q), by giving the suitable bn's according to a particular identity. Thus, we detected patterns for these identities while confirming what we found in our research.

For example, we made a list for the number of one distinct parts for integers starting from 1 up to 20:

```
→       onedist:[1,1,2,2,3,4,5,6,8,10,12,15,18,22,27,32,38,46,54,64];
(onedist) [1,1,2,2,3,4,5,6,8,10,12,15,18,22,27,32,38,46,54,64]
```

Then, we put this list as an input to our algorithm:

```
(%i11) f(onedist);
       [1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0]
(%o11) done
```

It can be easily realized that, the list showed up created a pattern as: for odd indexes the element is 1 and for even indexes it is 0. Concluding, one distinct parts formed a generating function with a list of exponents with odd parts. This confirmed the Euler identity which is discussed in part 2.1.

$$p(n \mid \text{odd parts}) = p(n \mid \text{distinct parts})$$

Another list we gave as an input to our code was 2-distinct parts.

```
→       twodist:[1,1,1,2,2,3,3,4,5,6,7,9,10,12,14,17,19,23,26,31];
(twodist) [1,1,1,2,2,3,3,4,5,6,7,9,10,12,14,17,19,23,26,31]
```

```
(%i14) f(twodist);
[1,0,0,1,0,1,0,0,1,0,1,0,0,1,0,1,0,0,1,0]
(%o14) done
```

The pattern appeared was, according to mod 5, 1 and 4'th indexes showed up as 1. Thus, we showed another identity mentioned in the book (Andrews and Eriksson):

$$p(n \mid \text{parts} \equiv 1 \text{ or } 4 \pmod 5) = p(n \mid \text{2-distinct parts})$$

Later on, we tried these identities listed below:

a. Schur's Partition Identity
b. Siladic's Theorem
c. Roger-Ramanujan Gordon
d. Capparelli's Identity

a. Roger-Ramanujan Gordon

Theorem: Let a and k be integers with $0 < a \leq k$. Let $A_{k,a}(n)$ denote the number of partitions of n into parts not of the forms $(2k+1)m$, $(2k+1)m\pm a$; $A_{k,a}(0)=1$. Let $B_{k,a}(n)$ denote the number of partitions of n of the form

$$n= b_1 + \ldots + b_n$$

with $b_i \geq b_{i+1}$, $b_i - b_{i+k-1} \geq 2$, and with 1 appearing as a summand at most a-1 times; $B_{k,a}(0)=1$. Then

$$A_{k,a}(n) = B_{k,a}(n)$$

for k=4 and a=3,

```
rorago(t):=   block( w:[],
   for n:1 thru t do(s:integer_partitions(n),
                x:[],
                k:4,
                a:3,
                for ptn in s do(
                   d:true,
                   if length(ptn)>1 then(
                   counter:k,
                   onecounter:0,
                   for el in ptn do(
                      if el=1 then onecounter: onecounter+1,
                      if onecounter=a or onecounter>a then d:false,
                      if counter<=length(ptn) then(
                      if el-ptn[counter]<2 then d:false),
                      counter:counter+1)),

                   if d=true then x: append(x,[ptn])),
      w: append(w,[length(x)])
      ),disp(w));
```

With the algorithm for this identity, we have created a list for the first 20 elements starting from 1. Then we put this list as an input to the previously implemented Kanade - Russell Theorem's code. What we found was a pattern of two 1's and one 0 repetitively.

```
rorago(20);
[1,2,2,4,5,7,9,13,16,22,27,36,44,57,70,89,108,135,163,202]
done

f([1,2,2,4,5,7,9,13,16,22,27,36,44,57,70,89,108,135,163,202]);
[1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1]
```

b. Siladic's Theorem

Theorem: The number of partitions $\lambda_1 + ... + \lambda_s$ of an integer n into parts different from 2 such that difference between two consecutive parts is at least 5 ($\lambda_i - \lambda_{i+1} \geq 5$) and

$$\begin{aligned}
\lambda_i - \lambda_{i+1} = 5 &\Rightarrow \lambda_i + \lambda_{i+1} \not\equiv \pm 1 \ \pm 5 \ \pm 7 \bmod 16,\\
\lambda_i - \lambda_{i+1} = 6 &\Rightarrow \lambda_i + \lambda_{i+1} \not\equiv \pm 2 \ \pm 6 \quad \bmod 16,\\
\lambda_i - \lambda_{i+1} = 7 &\Rightarrow \lambda_i + \lambda_{i+1} \not\equiv \pm 3 \quad \bmod 16,\\
\lambda_i - \lambda_{i+1} = 8 &\Rightarrow \lambda_i + \lambda_{i+1} \not\equiv \pm 4 \quad \bmod 16,
\end{aligned}$$

We implemented this theorem as:

```
(%i22) sila(t):= block( w:[],
    for n:1 thru t do(s:integer_partitions(n),
            x:[],
            for ptn in s do(
              a:true,
              if length(ptn)>1 then(
              b:0,
              for el in ptn do(
                if  el−b<5 and −5<el−b then a:false,
                if b−el=5 and (mod(el+b,16)=1 or mod(el+b,16)=15 or mod(el+b,16)=5 or mod(el+b,16)=11 or mod(el+b,16)=7 or mod(el+b,16)=9) then a:false,
                if b−el=6 and (mod(el+b,16)=2 or mod(el+b,16)=14 or mod(el+b,16)=6 or mod(el+b,16)=10) then a:false,
                if b−el=7 and (mod(el+b,16)=3 or mod(el+b,16)=13) then a:false,
                if b−el=8 and (mod(el+b,16)=4 or mod(el+b,16)=12) then a:false,
                b:el)),

            if a=true then x: append(x,[ptn])),
      w: append(w,[length(x)])
      ),disp(w));
```

c. Schur's Partition Identity

Theorem: For any positive integer n, the number of partitions into parts = ±1 (mod 6) equals the number of partitions into 3-distinct parts where no consecutive multiples of 3 appear.

```
s(t):=   block( w:[],
   for n:0 thru t do(s:integer_partitions(n),
                x:[],
                for ptn in s do(
                   a:true,
                   b:-5,
                   for el in ptn do(
                      if el-b<3 and -3<el-b then a:false,
                      b:el),
                   for el in ptn do(
                      if mod(el,3)=0 and member(el+3,ptn) then a:false),

                if a=true then x: append(x,[ptn])),
         w: append(w,[length(x)])

   ),disp(w));
s(20);
[1,1,1,1,1,2,2,3,3,3,4,5,6,7,8,9,10,12,14,16,18]
```

After partitioning integers between 1 and 20 according to Schur, we put these as an input to the code for Kanade and Russell's theorem.

3.Conclusion and Future Work

By this research, we have learned partitioning integers, multiple integer partition identities, and coding on Maxima. With our codes for identities, we showed the validity of the theorems in a much shorter time compared to trying them manually with writing.

Our future work will be trying to find our unique identities by trying more theorems on our code while seeking for patterns.

References

Andrews, G. (1966). An Analytic Proof of the Rogers-Ramanujan-Gordon Identities. *American Journal of Mathematics, 88*(4), 844-846. doi:10.2307/2373082

Dousse, J. (2014). A combinatorial proof and refinement of a partition identity of Siladić. *European Journal of Combinatorics*, *39*, 223-232. doi:10.1016/j.ejc.2014.01.008

Andrews, G. E., & Eriksson, K. (2004). Integer partitions. Cambridge, UK: Cambridge University Press.

Kanade, S. & Russell, M. C.. (2014). Identity Finder and Some New Identities of Rogers–Ramanujan Type. *Experimental Mathematics*. 24 (10), doi:1080/10586458.2015.1015186.