Enumeration Algorithms for *Restricted and Unrestricted* Compositions and Words

Daniel Page

April 8, 2011

Enumeration?

Two very different disciplines

- Counting
- Generation

Two different disciplines in computation but a generation solution is more powerful.

Typical interests:

- **Combinatorial Objects**
- **Discrete Solution Spaces**
- **Sequencing**
- **Many more...**

Enumeration?

Primary Goals of an enumeration algorithm:

- Given properties of a finite set, exhaust all the members of the set which fit the definition.
- Avoiding extra structure whenever possible.
- Understanding more about the structure of a construction.
- Can solve counting problems.
- Eg) Combinatorial Objects, Word Problems...

WARNING: These algorithms are highly output sensitive.

Combinatorial Composition

Definition: A finite set $C_{s,n}$ which contains all non-negative integer sequences $л = л_1\, л_2 \ldots л_n$ such that $\sum (л_i) = s$, $1 \leq i \leq n$. This is called an *unrestricted composition*.

eg) $C_{3,3}$= {(0,0,3),(0,3,0),(3,0,0),

(2,1,0),(1,2,0),(0,2,1),(0,1,2),(1,0,2),(2,0,1),
(1,1,1)}

Combinatorial Composition

There is also a more general version:

A finite set $C^R_{s,n}$ which contains all non-negative integer sequences $л = л_1 л_2 \ldots л_n$ such that $\sum (л_i) = s$, $1 \leq i \leq n$, **where $л_i$ ϵ R**.

This is called an *restricted composition*. There are several types of restricted compositions…

eg) $C^R_{3,3} = \{(1,1,1)\}$, where $R = \{1\}$

History

Compositions play a vital role in the foundations of Theoretical Computer Science and also are of interest in Combinatorics.

Generation of Compositions are a specific sub-type of Word Problems.

Word Problem – Given a semi-Thue System,

- rewrite a string given a set of 'rewriting' axioms, to produce an outputting string in the string rewriting system in a language.
- General Case: Problem is *undecidable.*

History

Early Research:

Percy Alexander MacMahon (1854-1929)

"Compositions are merely partitions in which the order of occurrence of the parts are important"

- Many of his conventions are still used today.
- Earliest results.
- *"Assemblage of objects"*

Credit: "Combinatorics of Compositions and Words", Heubach,Mansour

History

Early Research:

Axel Thue (1863-1922)

"Zeichenreihen"

- First to systematically study words and compositions.
- Extended to the infinite cases. The general word problem.
- Thue and semi-Thue systems.
- *"Combinatorics on Words"*

Credit: "Combinatorics of Compositions and Words", Heubach,Mansour

Problems

Problem 1:

Given positive integers n,s (n =0 iff s = 0), enumerate the unrestricted composition $C_{s,n}$.

- Some Results:
- C(s,n) = CHOOSE(s+n-1,n-1)
- Lower bound space-complexity: Ω(C(s,n)).
  - Output sensitive
  - Tight bound (it's the output itself).

Problems

<u>Output Sensitivity Extremes</u>:

Keep this in mind! This is your solution space size that you output…

- C(8,7)= 3003.
- C(9,9) = 24310.
- C(200,100) = 1.38608382108618824826 11278421088 x 10^{81}

Problems

Some Solutions to Problem 1:

Often solutions try to take one element and form the next from the previous to avoid consuming extra space.

Randomized Generation

- Idea: Keep generating unique sequences of unique sum of s from a uniform distribution on [0,s] and test if it's currently in the set.
- Hard to analyze, but often useful in parallel implementations.

Problems

Some Solutions to Problem 1:

K-bit Reflected Gray Code Generation:

- Idea: Generate all gray codes in lexicographical order of length n integers in binary, then on the last iteration, remove all elements which don't sum to s if the last symbol is added.
- Constructs elements from scratch but the trade off could be dependent on implementation. (moves to modern results)
- Bitner-Ehrlich-Reingold (BER) method.

Problems

Some Solutions to Problem 1:

Brute Force:

- Idea: Similar to the previous technique except working directly with integers.
- Based on the definition, this technique can always take advantage of the fact we have a continuous interval of non-negative integers.
- Simpler to implement but space can grow substantially more than the previous. Very small instances can be efficient.

Problems

Upper Bounds/Lower Bounds:

This problem has been attacked numerous times by many researchers since the 70's by the likes of Knuth, Lothaire, and many others.

Optimal solution: Combinatorial Gray Codes.

For simplicity: let k be the output size of a given composition.

Problems

Vital Result:

Goal: Compute gray codes quickly, and perform the 2nd technique (with modification).

- O(1) – Generating gray codes in O(1) worst-case time per word [Walsh,2003]
- Map this problem to our unrestricted composition problem
- Upper Bound – Ω(k) [evident from above]
- Optimal Time.

Problems

Problem 2 (Definition):

A finite set $C^{L,U}_{s,n}$ which contains all integer sequences $л = л_1\ л_2 \ldots л_n$ such that $\sum (л_i) = s$, $1 \le i \le n$, **where $L \le л_i \le U$**.

This is called an *bounded composition*.

- *Specific type of restricted composition.*
- Inherits problems from problem 1.
- New problems: unused space and decision complexity increases.

Problems

Problem 2:

Given positive integers n,s (n =0 iff s = 0), and non-negative integers L,U, enumerate the bounded composition $C^{L,U}_{s,n}$.

- *Why would we want this?*
  - *Many of the same reasons, but some of the more modern research was for statistical analysis in computation.*
- Applying the last problems solutions can be tedious but are doable but can lead to space-issues in contrast to solution size.

Problems

<u>Problem 2</u>:

Using the same techniques as Problem 1 would cause a rise in time-complexity using the optimal solution.

- When L=0 and U=s, we get the unrestricted composition.
- Same lower bound holds since this is a sub-instance of Problem 1.

Problems

Some Solutions to Problem 2:

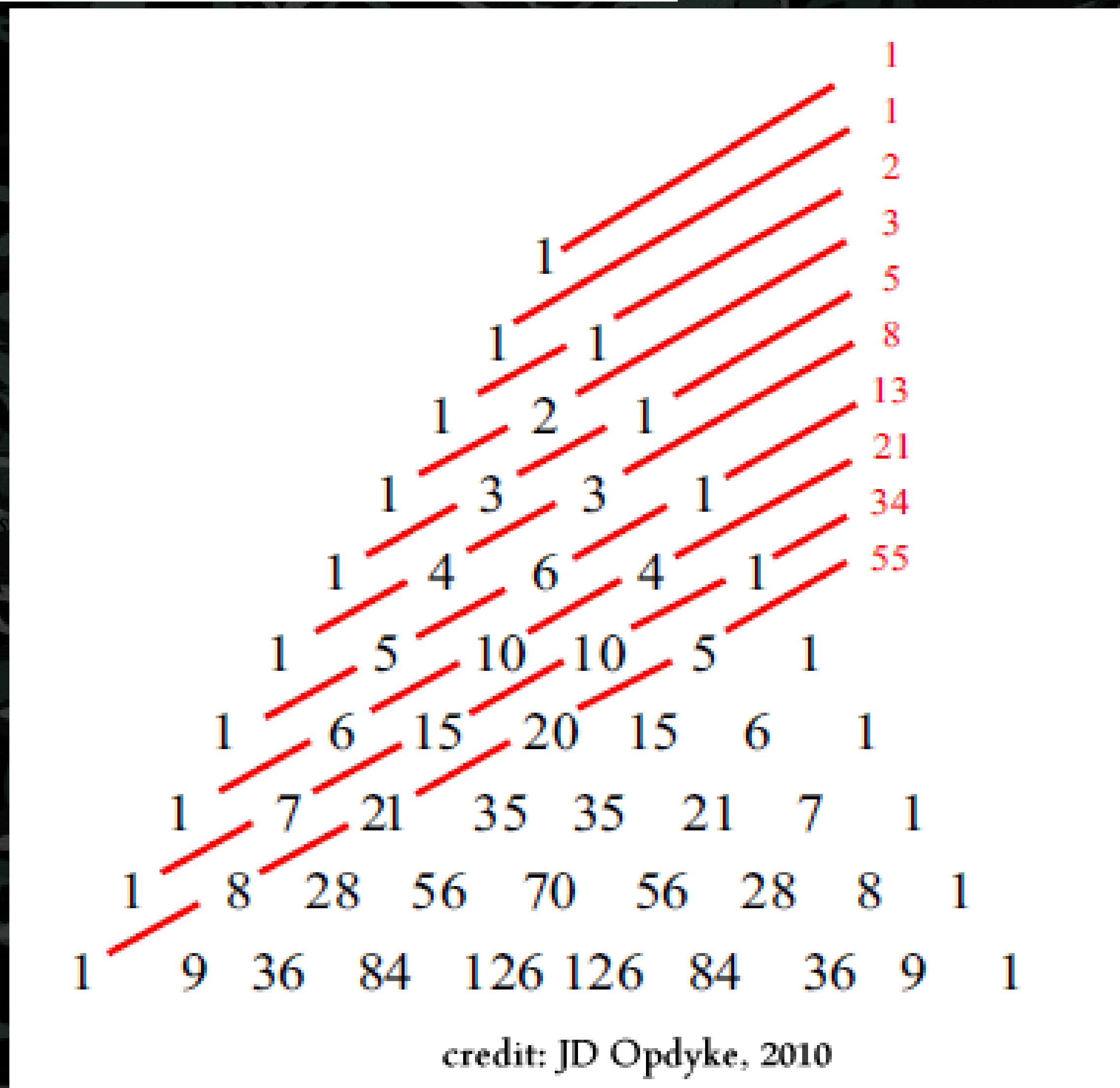
Another technique:

Idea: Based on Fibonnaci Numbers and Pascal's Triangle for counting.[Kimberling. 2002]

- Inside pascal's triangle there exists a path from an interior node to the root that will diagonally reach the root of the triangle. Thus when reading off the coefficients of the triangle you obtain a composition sequence of elements for each off diagonal.
- Picture... next here...

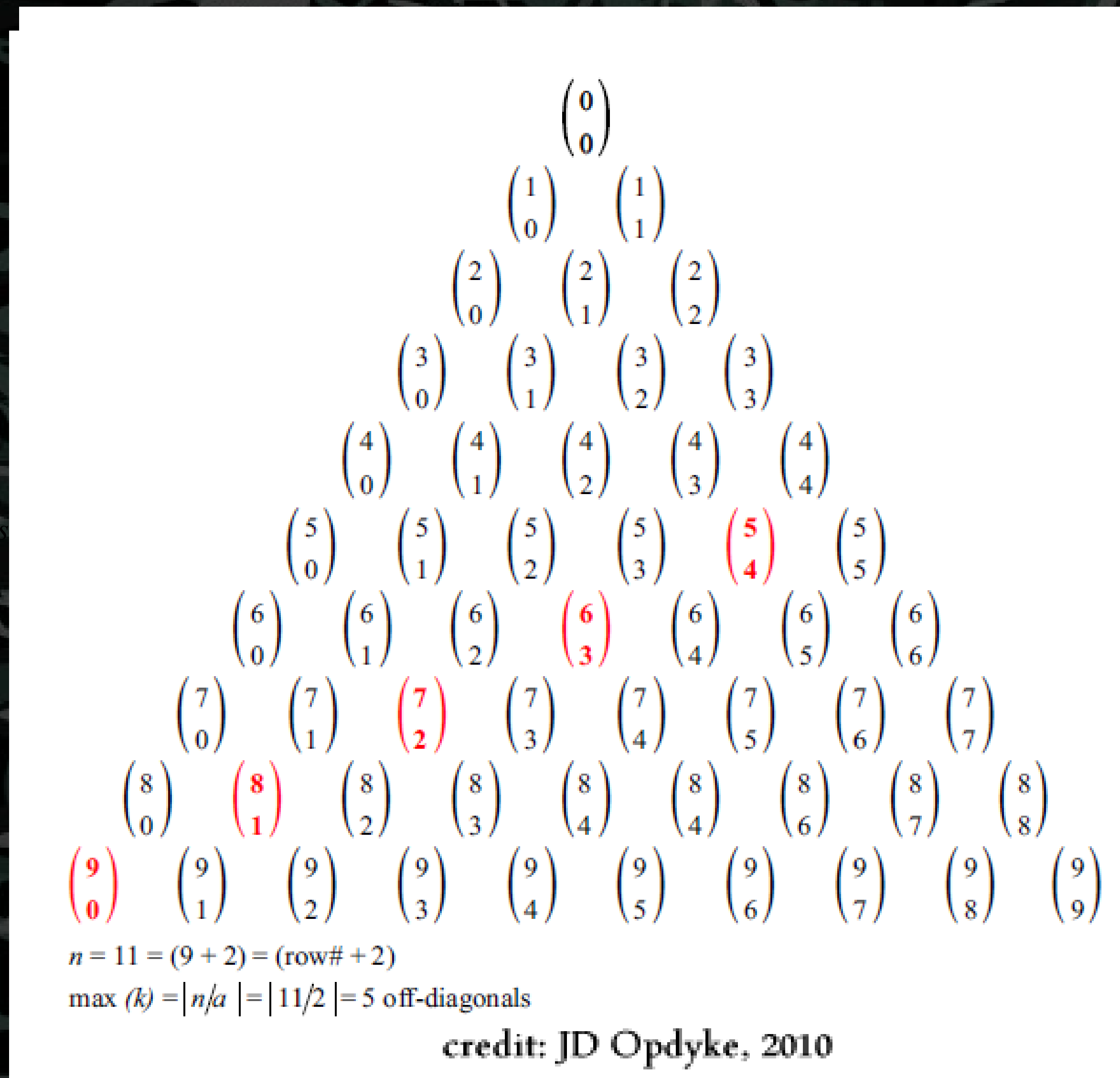
Problems

Some Solutions to Problem 2:



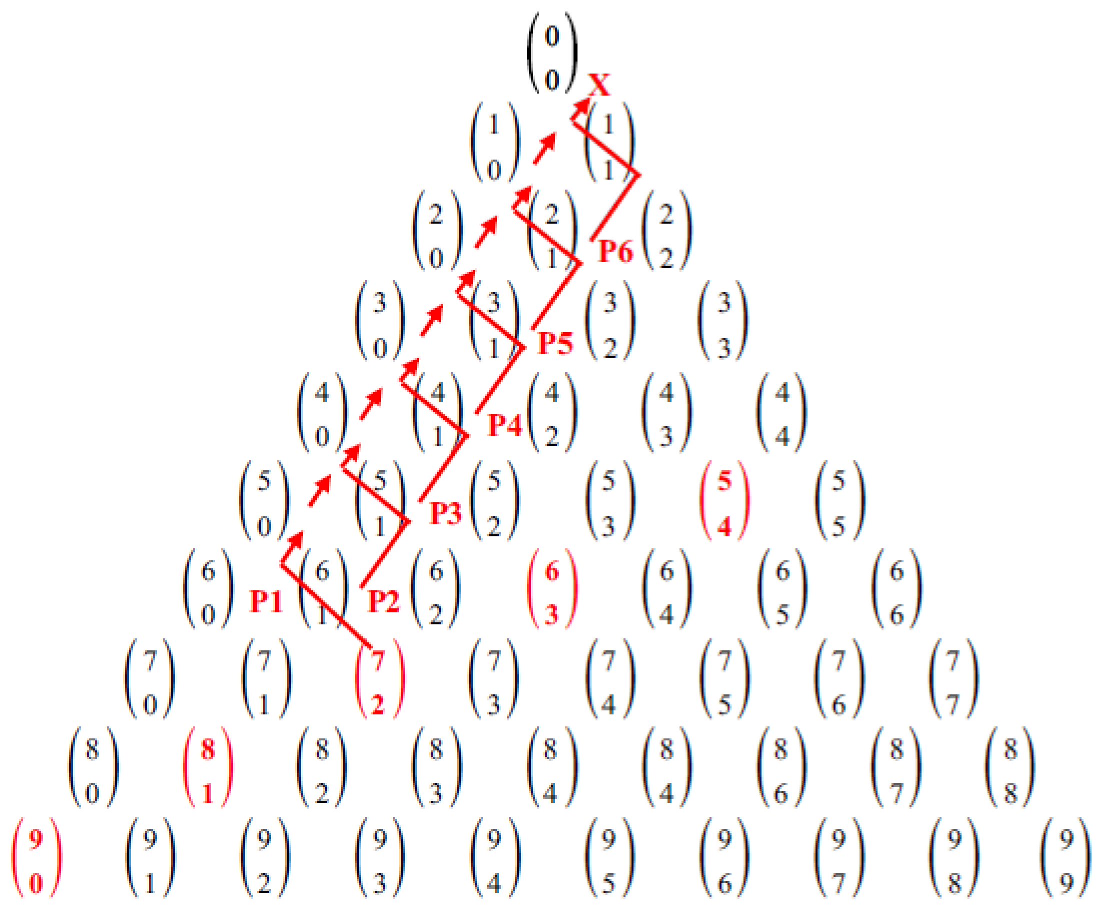
Problems

Some Solutions to Problem 2:



Problems

Some Solutions to Problem 2:



Composition paths traced by RICs_Base($n = 11$, $k = 3$)

credit: JD Opdyke, 2010

Problems

Some Solutions to Problem 2:

This solution was developed by J.D. Opdyke in 2010 from Kimberling's result.

$$O\left(\sum_{k=\min k}^{\max k}\left(k \cdot \sum_{i=(n-b)}^{(n-a)} c\,(i, k-1, a, b)\right)\right)$$

- time-complexity: ~O(k x RICs)
- Hard to bound the algorithm's run-time.
- Flaws: Recursive, and has loops.
- Is the best result in Bounded Compositions.

Future Problems

Problem 3:

Given positive integers n,s (n =0 iff s = 0), and set of non-negative integers R, enumerate the restricted composition $C^R_{s,n}$.

- Very tough problem. General 'first order'.
- Has reduction to Positive Integer Subset Sum
- Problem 1 and Problem 2's solutions won't work on this without a heavy cost. Discontinuity of values.
- Work on counting has been done for this but, not generating.

Future Research

Research Interests:

We had mentioned Problem 3 but what other interests hold in this discourse with Compositions still?

- Pattern avoidance work, w-avoidance
- Relationships between Integer Partitions and Integer Compositions
- '2nd order' problems with compositions.

Questions?

Thank you very much for your time!

$s+k-1, n$

Questions?

Have a beautiful day!