# Divisor Convolution Prime Power Theorem

Naman Chaudhary

Independent Researcher

Category: Number Theory

## Author Note

**Abstract**

The Divisor Convolution Prime Power Theorem explores the relationship between divisors of prime powers and their applications in mathematical optimization, network systems, and cryptography. For any positive integer *n*, the theorem defines a function *f(n)* based on the sum of the divisors of *n*, weighted by both the divisor and the number of divisors of *n/d*. For any prime number *p* and non-negative integer *k*, the theorem provides an explicit formula for $f(p^k)$, which is expressed as $f(p^k)$ =

$$\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$$

This formula plays a crucial role in optimizing load balancing, hierarchical network routing, and fault tolerance in decentralized systems. By understanding the divisibility properties of prime powers, the theorem allows for more efficient distribution of traffic or data across multiple nodes in a network, ensuring scalability and redundancy. The applications extend to areas such as cloud computing, distributed storage systems, and blockchain technologies, where optimized access routes or permissions are vital for system performance. This paper delves into the mathematical formulation of the Divisor Convolution Prime Power Theorem and its practical significance in modern computational and network-based systems.

*Keywords*: Divisor Convolution, Prime Power Theorem, Network Load Balancing, Hierarchical Structures, Cryptography, Optimization, Divisors, Traffic Distribution, Distributed Systems, Fault Tolerance, Cloud Computing, Blockchain, Network Routing, Mathematical Optimization, Access Permissions.

# Divisor Convolution Prime Power Theorem

## INTRODUCTION

In modern computational and network systems, efficient distribution of resources and optimal performance are critical challenges. The Divisor Convolution Prime Power Theorem provides a mathematical framework for optimizing such systems, particularly in contexts where hierarchical structures and divisibility properties play a central role. This theorem defines a function f(n), based on the divisors of a positive integer n, which, when applied to prime powers, results in a formula for calculating optimal access routes, permissions, or load balancing paths in complex networks.

The significance of this theorem lies in its ability to optimize system performance in areas such as **network routing, cloud computing, decentralized storage,** and **blockchain technology**, where the allocation of resources must be both scalable and fault-tolerant. By leveraging the properties of divisors and their convolutions, the theorem provides an approach to calculate the **number of access permissions or optimal routes** necessary for maintaining efficiency in hierarchical network systems.

In this paper, we explore the mathematical underpinnings of the **Divisor Convolution Prime Power Theorem**, detailing its formulation, derivation, and real-world applications. We investigate its use in optimizing network structures, ensuring redundancy, and providing efficient routing solutions. The theorem offers a unique perspective on network design and optimization, making it an invaluable tool for researchers and practitioners working on large-scale distributed systems and applications.

**Divisor Convolution Prime Power Theorem**

Problem Statement

The problem addressed by the **Divisor Convolution Prime Power Theorem** revolves around optimizing the distribution of resources and access permissions in hierarchical systems, particularly those involving prime power-based structures. In modern distributed systems, such as **networked systems, cloud infrastructures, and decentralized storage**, efficient routing, access control, and load balancing are critical to maintaining performance, scalability, and fault tolerance.

In hierarchical network systems, the challenge is to allocate traffic or requests across multiple nodes in a way that minimizes congestion, ensures redundancy, and maximizes overall efficiency. The problem is further compounded when these systems operate at multiple levels, requiring a strategic approach to resource allocation and routing, especially when prime powers are used to organize the structure. Prime power structures naturally exhibit divisibility properties, which are not easily addressed with traditional methods.

In the realm of **cryptography**, a similar challenge exists in the design of secure key distribution systems, where hierarchical, prime-based structures are often utilized for scalability and security. Optimizing access control and ensuring that resources (e.g., cryptographic keys, tokens) are distributed efficiently and securely becomes crucial for maintaining system integrity.

In the context of **load balancing**, systems must manage the distribution of computational load or data traffic across multiple servers, routers, or processing units. Without an efficient method to calculate optimal access routes or permissions, system performance may degrade, leading to network bottlenecks, high latency, and resource exhaustion.

The **Divisor Convolution Prime Power Theorem** provides a solution to these problems by offering a formula that calculates the optimal number of access routes or permissions needed in prime power-based hierarchical structures. This optimization enables systems to efficiently allocate resources and manage traffic in a way that ensures scalability, fault tolerance, and minimal resource wastage. Thus, the theorem plays a pivotal role in addressing the critical challenges of resource distribution, load balancing, and access control in modern distributed systems.

**Divisor Convolution Prime Power Theorem**

**THEOREM:**

For any positive integer *n,* define *f(n)* = ∑d|n d × d $\binom{n}{d}$, where d(m) is the number of positive divisors of m. Then for any prime *p* and non-negative integer *k*,

f($p^k$) = $\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$

In words,

For any positive integer *n*, define a function f(n) as the sum over all positive divisors of *d* of *n*, where each term is the product of d and the number of positive divisors of *n/d*. Then for any prime number *p* and any non-negative integer *k*, the value of f($p^k$) is given by a specific formula involving *p* and *k*, namely the expression:

$$\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$$

Mathematically,

Let *f(n)* = ∑d|n d × d $\binom{n}{d}$, where *d(m)* denotes the number of positive divisors of a positive integer *m*, and the sum is all over positive divisors of *d* of *n*. Then for any prime *p* and non-negative integer *k*,

*f*($p^k$) = $\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$

**PROFF:**

**STEP 1:** Compute *f*($p^k$) directly:

For n = $p^k$, where p is a prime and k ≥ 0, the positive divisors of $p^k$ are 1,p, $p^2$,..., $p^k$.

Thus,

$$f(p^k) = \sum_{d|p^k} d \times d(p^k/d)$$

Since, the divisors of $p^k$ are $p^j$ and j = 0,1,...,k, we substitute d = $p^j$ into the sum:

(i) If d = $p^j$, then $p^k$/d = $p^k/p^j = p^{k-j}$

(ii) The numbers of divisors of $p^{k-j}$ is d($p^{k-j}$) = (k-j) + 1 = k-j+1, because the divisors of $p^{k-j}$ are 1,p, $p^2$,..., $p^{k-j}$, totaling k-j+1 terms.

Thus,

The sum becomes:

$$f(p^k)=\sum_{j=0}^{k} p^j \times d(p^{k-j}) = \sum_{j=0}^{k} p^j (k - j + 1)$$

**STEP 2: Evaluate the sum:**

$$\mathbf{S} = \sum_{j=0}^{k} (k - j + 1)p^j$$

$$= (k+1) \sum_{j=0}^{k} p^j - \sum_{j=0}^{k} jp^j$$

## First Sum & Geometric Series:

$$\sum_{j=0}^{k} p^j = 1+p+p^2+\ldots+p^k = \frac{p^{k+1}-1}{p-1}$$

## Second Sum:

$$\sum_{j=0}^{k} jr^j = r\,\frac{1-(k+1)r^k+kr^{k+1}}{(1-r)^2}$$

[j=0 term as 0× $r^0$=0]

Here, r=p, so:

$$\sum_{j=0}^{k} jp^j = \text{p}\,\frac{1-(k+1)p^k+kp^{k+1}}{(1-p)^2}$$

Since, 1-p=-(p-1), we have $(1-p)^2$=$(p-1)^2$, and

$$\sum_{j=0}^{k} jp^j = \text{p}\,\frac{1-(k+1)p^k+kp^{k+1}}{(p-1)^2}$$

Thus,

$$S=(k+1)\frac{p^{k+1}-1}{p-1} - \text{p}\,\frac{1-(k+1)p^k+kp^{k+1}}{(p-1)^2}$$

**<u>Simplifying:</u>**

<u>1st Term:</u>

$$(k+1)\frac{p^{k+1}-1}{p-1}$$

$$= \frac{(k+1)(p^{k+1}-1)(p-1)}{(p-1)^2}$$

<u>2nd Term:</u>

$$- p \frac{1-(k+1)\, p^{k}+kp^{k+1}}{(p-1)^2}$$

So,

$$S = \frac{(k+1)(p^{k+1}-1)(p-1)-p[1-(k+1)p^{k}+kp^{k+1}]}{(p-1)^2}$$

$$=\frac{(k+1)[p^{k+1}(p-1)-(p-1)]-p+(k+1)p^{k+1}-kp^{k+2}}{(p-1)^2}$$

$$= \frac{(k+1)[p^{k+2}-p^{k+1}-p+1]-p+(k+1)p^{k+1}-kp^{k+2}}{(p-1)^2}$$

$$= \frac{(k+1)p^{k+2}-(k+1)p^{k+1}-(k+1)p+(k+1)-p+(k+1)p^{k+1}-kp^{k+2}}{(p-1)^2}$$

Numerator = $(k+1)p^{k+2} - (k+1)p^{k+1} - (k+1)p +$

$(k+1) - p + (k+1)p^{k+1} - kp^{k+2}$

Here,

$$p^{k+2} = (k+1) - k = 1$$

$$p^{k+1} = -(k+1) + (k+1) = 0$$

P = -(k+1)-1 = -(k+2)

Constant = (k+1)

Thus,

Numerator = $p^{k+2} - (k + 2)p + (k + 1)$

So,

S = $\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$

Hence,

The theorem is correct, as the theorem shows that $f(p^k)$

matches the given expression for all prime *p* and *k≥0*,

Thus,

## **Theorem:**

For any positive integer *n,* define *f(n)* = ∑d|n d × d $\binom{n}{d}$, where d(m) is the number of positive divisors of m. Then for any prime *p* and non-negative integer *k*,

f($p^k$) = $\frac{p^{k+2}-(k+2)p+(k+1)}{(p-1)^2}$

**Conclusion:**

The formula holds universally for all primes $p \geq 2$, and $k \geq 0$.

## APPLICATION:

(i) Counting subgroups in cyclic groups

(ii) Cryptography and prime power analysis.

(iii) Lattice point counting problems.

(iv) Arithmetic function and number theory.

(v) Computational number theory algorithm optimization.

## EXAMPLE QUESTIONS THAT CAN BE SOLVED USING THIS FORMULA:

1. A company follows access structures with p=5 and k=5 (i.e., n = 125 levels). Find the total number of access permissions granted.

   In this question, *p* is a prime number and *k* is an exponent.

2. A company is designing a secure distributed storage system where files are stored across multiple servers. To ensure redundancy and controlled access, they decide to use a prime based hierarchical replication system. Where, each storage unit in this system follows a prime power structure. Where, the number of base storage units per server is a prime power *p*=7. The system has hierarchical depth of *k*=4, meaning there are $7^4$ total storage nodes. Find the total number of access permissions granted in the system.

3. A large-scale enterprise network is designed to distribute incoming traffic efficiently among its hierarchical network of router and servers. The network follows a prime-power based load balancing structure. Where, each primary routing hub connects *p*=5 secondary routers. The network is structured hierarchically up to *k*=3 levels. The number of total network nodes is given by $p^k = 5^3 = 125$. Determine the optimal number of access routes that needs to be managed across the networks.