**เสริมเนื้อหาบทที่ 8 การคำนวณแบบขนาน (Parallel Computing)ด้วยบอร์ด Pi3/Pi4**

เนื่องด้วยจากกฎของ Amdahl สามารถประยุกต์กับการคำนวณค่า Speedup ของการคูณเมทริกซ์แบบขนานนี้ได้จึงได้เลือกเพิ่มเนื้อหาในส่วน กฎของ Amdahl โดยมีเนื้อหาดังนี้

**กฎของ Amdahl(Amdahl's law)**

เป็นสูตรที่ทำให้ทฤษฎี[เพิ่มความเร็ว](https://hmong.in.th/wiki/Speedup)ใน[ความล่าช้า](https://hmong.in.th/wiki/Latency_(engineering))ของการดำเนินการของงานที่ได้รับการแก้ไขเป็น[ภาระงาน](https://hmong.in.th/wiki/Workload)ที่สามารถคาดหวังของระบบที่มีทรัพยากรที่ดีขึ้น โดยตั้งชื่อตาม[นักวิทยาศาสตร์คอมพิวเตอร์](https://hmong.in.th/wiki/Computer_scientist) [Gene Amdahl](https://hmong.in.th/wiki/Gene_Amdahl) และถูกนำเสนอในการประชุม [AFIPS](https://hmong.in.th/wiki/American_Federation_of_Information_Processing_Societies) Spring Joint Computer Conference ในปี 1967 ซึ่งกฎของ Amdahl มักใช้ในการ[คำนวณแบบคู่ขนาน](https://hmong.in.th/wiki/Parallel_computing)เพื่อหาความเร็วตามทฤษฎีเมื่อใช้โปรเซสเซอร์หลายตัว

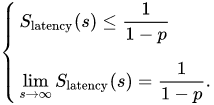
**คำนิยาม (Definition)**

คำจำกัดความ: “ การปรับปรุงที่ได้รับในประสิทธิภาพของระบบเนื่องจากการสับเปลี่ยนของส่วนประกอบอย่างใดอย่างหนึ่งนั้นถูก จำกัด ด้วยส่วนของเวลาที่ใช้ส่วนประกอบ”

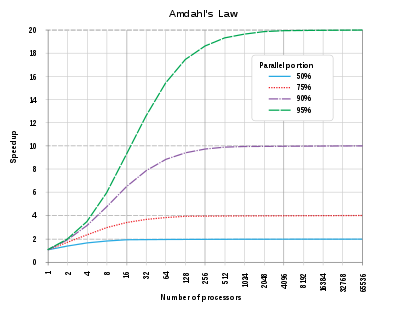


* **S latency** เป็น Speedup ตามทฤษฎีของการปฏิบัติงานทั้งหมด
* **s** คือ Speedup ของส่วนหนึ่งของงานที่ได้รับประโยชน์จากทรัพยากรระบบที่ได้รับการปรับปรุง
* **p** คือสัดส่วนของเวลาดำเนินการที่ส่วนที่ได้รับประโยชน์จากทรัพยากรที่ปรับปรุงแล้วซึ่งเดิมครอบครองอยู่

นอกจากนี้



แสดงให้เห็นว่า Speedup ตามทฤษฎีของการดำเนินงานทั้งหมดเพิ่มขึ้นด้วยการปรับปรุงทรัพยากรของระบบและโดยไม่คำนึงถึงขนาดของการปรับปรุง Speedup ตามทฤษฎีมักจะถูกจำกัดโดยส่วนของงานที่ไม่ได้รับประโยชน์จากการปรับปรุง



ที่มา : [Amdahl's law - Wikipedia](https://en.wikipedia.org/wiki/Amdahl%27s_law)

การเพิ่มความเร็วตามทฤษฎีของเวลาแฝงของการดำเนินการโปรแกรมตามฟังก์ชันของจำนวนโปรเซสเซอร์ที่ดำเนินการ ตามกฎของ Amdahl การเร่งความเร็วจะถูกจำกัดโดยส่วนอนุกรมของโปรแกรม ตัวอย่างเช่น หาก 95% ของโปรแกรมสามารถขนานกันได้ ความเร็วสูงสุดตามทฤษฎีโดยใช้การคำนวณแบบขนานจะเท่ากับ 20 เท่า

**ที่มาของสูตร :**

งานที่ดำเนินการโดยระบบซึ่งมีการปรับปรุงทรัพยากรเมื่อเปรียบเทียบกับระบบเริ่มต้นที่คล้ายคลึงกันสามารถแบ่งออกเป็นสองส่วน :

* ส่วนที่ไม่ได้รับประโยชน์จากการปรับปรุงทรัพยากรของระบบ
* ส่วนที่ได้รับประโยชน์จากการปรับปรุงทรัพยากรของระบบ

เวลาการดำเนินการของงานทั้งหมดก่อนการปรับปรุงทรัพยากรของระบบจะแสดงเป็น  **T** รวมถึงเวลาดำเนินการของส่วนที่จะไม่ได้รับประโยชน์จากการปรับปรุงทรัพยากรและเวลาดำเนินการในส่วนที่จะได้รับประโยชน์ เศษส่วนของเวลาดำเนินการของงานที่จะได้รับประโยชน์จากการปรับปรุงทรัพยากรแสดงโดย **p** ส่วนที่เกี่ยวกับส่วนที่ไม่ได้รับประโยชน์จากสิ่งนั้นก็คือ **1-p** จากนั้น :

****

การดำเนินการในส่วนที่ได้รับประโยชน์จากการปรับปรุงทรัพยากรที่เร่งโดยปัจจัย **s** หลังจากการปรับปรุงทรัพยากร ดังนั้น เวลาดำเนินการของส่วนที่ไม่ได้ประโยชน์จากมันยังคงเท่าเดิม ในขณะที่ส่วนที่ได้ประโยชน์จะกลายเป็น :

เวลาดำเนินการตามทฤษฎี **T(s)** ของงานทั้งหมดหลังจากการปรับปรุงทรัพยากรแล้ว :



กฎของ Amdahl ให้[ความเร็ว](https://hmong.in.th/wiki/Speedup)ทางทฤษฎีใน[ความหน่วงแฝง](https://hmong.in.th/wiki/Latency_(engineering))ของการดำเนินการงานทั้งหมดที่ปริมาณงานคงที่(**W**) ซึ่งส่งผลให้ :



**การประยุกต์ใช้กับ Speedup**

Speedup คือการถูกจำกัดโดยเวลาทั้งหมดที่จำเป็นสำหรับส่วนต่อเนื่อง (serial) ของโปรแกรม สำหรับการคำนวณ 10 ชั่วโมง หากเราสามารถทำการการคำนวณ 9 ชั่วโมงแบบขนานและ 1 ชั่วโมงไม่ใช่แบบขนานกันได้ ความเร็วสูงสุดของเราจะถูกจำกัดให้เร็วขึ้น 10 เท่า หากคอมพิวเตอร์เร็วขึ้น ตัวเร่งความเร็วก็จะเท่าเดิม

ถ้าใช้โปรเซสเซอร์ n ตัว Speedup เป็น :

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ

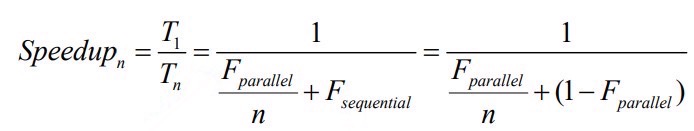
ที่ T1 คือเวลาดำเนินการในแกนเดียว, Tn คือเวลาดำเนินการบน n cores, Speedup ควรจะ >1

Speedup Efficiency คือ

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ

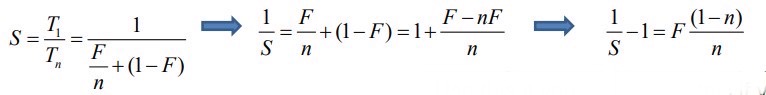
ซึ่ง Amdahl's Law บอกว่า

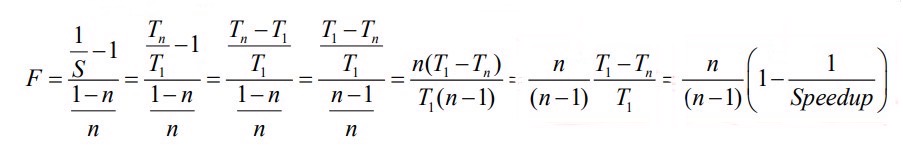


F parallel / n : เศษส่วนนี้สามารถลดลงได้โดยการใช้โปรเซสเซอร์หลายตัว, F sequential : ไม่สามารถลดลงได้

**สามารถหา F parallel ได้โดยใช้ Amdahl's Law เมื่อรู้ speedup และ จำนวนของโปรเซสเซอร์**

Amdahl's Law บอกว่า **:**

****

****

หากค่า (n,S) มีหลายค่า สามารถใช้ค่าเฉลี่ย (โดยความจริงแล้วเป็นกำลังสองที่น้อยที่สุด) :

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ ถ้า i = 1, T ni = T1

**กฎของ Amdahl สามารถให้ความเร็วสูงสุดได้**

เศษส่วนดังต่อไปนี้กำหนดขอบเขตบนว่าจะได้รับประโยชน์มากหรือน้อยเท่าใดจากการเพิ่มตัวประมวลผลมากขึ้น :

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ

|  |  |
| --- | --- |
| F parallel | maxSpeedup |
| 0.00 | 1.00 |
| 0.10 | 1.11 |
| 0.20 | 1.25 |
| 0.30 | 1.43 |
| 0.40 | 1.67 |
| 0.50 | 2.00 |
| 0.60 | 2.50 |
| 0.70 | 3.33 |
| 0.80 | 5.00 |
| 0.90 | 10.00 |
| 0.95 | 20.00 |
| 0.99 | 100.00 |

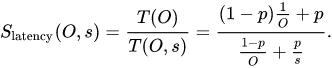
ตัวอย่างเช่น หากโปรแกรมต้องใช้เวลา 20 ชั่วโมงจึงจะเสร็จสมบูรณ์โดยใช้เธรดเดียว แต่ส่วนหนึ่งชั่วโมงของโปรแกรมไม่สามารถขนานกันได้ ดังนั้นเวลาดำเนินการที่เหลือเพียง 19 ชั่วโมง ( p = 0.95 ) เท่านั้นที่จะสามารถขนานกันได้ จำนวนเธรดที่อุทิศให้กับการดำเนินการแบบขนานของโปรแกรมนี้ เวลาดำเนินการขั้นต่ำต้องไม่น้อยกว่า 1 ชั่วโมง ดังนั้นการเร่งความเร็วตามทฤษฎีจึงจำกัดไว้ที่ 20 เท่าของประสิทธิภาพของเธรดเดียว ( 1 / (1 – p) = 20 )

**การเพิ่มประสิทธิภาพส่วนต่อเนื่องของโปรแกรมคู่ขนาน**

ถ้าส่วนที่ไม่ขนานกันถูกปรับให้เหมาะสมโดยปัจจัยของ O

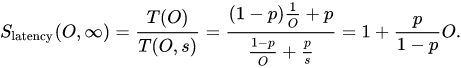


เป็นไปตามกฎของ Amdahl ที่ว่าการเร่งความเร็วเนื่องจากการขนานกันนั้นถูกกำหนดโดย

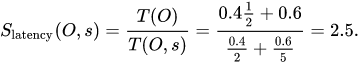


เมื่อ s = 1,จะมี ซึ่งหมายความว่าการเร่งความเร็วจะถูกวัดตามเวลาดำเนินการหลังจากปรับชิ้นส่วนที่ไม่เป็นแบบขนาน

เมื่อ s =  ∞



ถ้า 1 – p = 0.4 , O = 2 และ s = 5 แล้ว

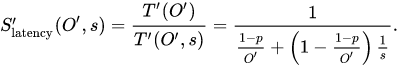


**การแปลงชิ้นส่วนที่ต่อเนื่องกันของโปรแกรมคู่ขนานให้เป็นแบบขนาน**

พิจารณากรณีที่ส่วนที่ไม่ขนานกันลดลงด้วยค่า O' และส่วนที่ขนานกันได้จะเพิ่มขึ้นตามลำดับ แล้ว



มันเป็นไปตามกฎของ Amdahl ที่ว่าการเร่งความเร็วเนื่องจากการขนานกันนั้นถูกกำหนดโดย



ที่มาข้างต้นสอดคล้องกับการวิเคราะห์ของ Jakob Jenkov เกี่ยวกับเวลาดำเนินการกับการแลกเปลี่ยนเพื่อเร่งความเร็ว

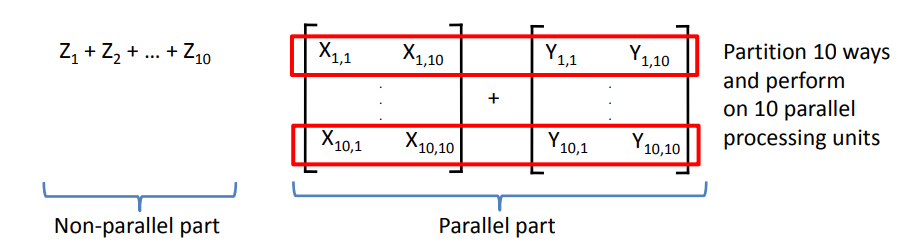
**ตัวอย่าง(Examples)**

สมมุติหาก 30% ของเวลาดำเนินการอาจเป็นเรื่องของการเพิ่มความเร็ว p จะเป็น 0.3; ถ้าต้องการปรับปรุงทำให้ส่วนที่ได้รับผลกระทบเร็วขึ้นสองเท่า s จะเป็น 2 กฎของ Amdahl ระบุว่าการเร่งความเร็วโดยรวมของการใช้การปรับปรุงจะเป็นดังนี้:

**S** latency = 1 / 1 - p + ( p / s ) = 1 / 1 – 0.3 + (0.3/2) = 1.18

สมมติว่าเราได้รับงานต่อเนื่องซึ่งแบ่งออกเป็นสี่ส่วนต่อเนื่องกัน ซึ่งมีเปอร์เซ็นต์ของเวลาในการดำเนินการคือ p 1 = 0.11 , p 2 = 0.18 , p 3 = 0.23 และ p 4 = 0.48 ตามลำดับ จากนั้นเราจะบอกว่าส่วนที่ 1 ไม่เร่งความเร็ว ดังนั้น s1 = 1ในขณะที่ส่วนที่ 2 เร่งขึ้น 5 เท่า ดังนั้น s2 = 5 ส่วนที่ 3 เร่งขึ้น 20 ครั้ง ดังนั้น s3 = 20 , และส่วนที่ 4 คือการเร่งความเร็วขึ้น 1.6 เท่า ดังนั้น s4 = 1.6 โดยใช้กฎของ Amdahl การเร่งความเร็วโดยรวมคือ

**S** latency = 1 / (p1/s1) + (p2/s2) + (p3/s3) + (p4/s4) = 1 / (0.11/1) + (0.18/5) + (0.23/20) + (0.48/1.6) = 2.19



ที่มา : [Microsoft PowerPoint - 16LecSu12TLP.pptx (berkeley.edu)](https://inst.eecs.berkeley.edu/~cs61c/su12/lec/16/16LecSu12TLP.pdf)

พิจารณา การรวมตัวแปร scalar 10 ตัวและ 10 x 10 เมทริกซ์ (ผลรวมเมทริกซ์) บนโปรเซสเซอร์ 10 ตัว

**S** latency = 1 / 1 - p + ( p / s ) = 1 / 1 - 0.909 + ( 0.909 / 10 ) = 5.5

ถ้าเป็นโปรเซสเซอร์ 100 ตัว

**S** latency = 1 / 1 - p + ( p / s ) = 1 / 1 - 0.909 + ( 0.909 / 100 ) = 10.0

ถ้าเป็น 100 x 100 เมทริกซ์ บนโปรเซสเซอร์ 10 ตัว

**S** latency = 1 / 1 - p + ( p / s ) = 1 / 1 - 0.999 + ( 0.999 / 10 ) = 9.9

ถ้าเป็น 100 x 100 เมทริกซ์ บนโปรเซสเซอร์ 100 ตัว

**S** latency = 1 / 1 - p + ( p / s ) = 1 / 1 - 0.999 + ( 0.999 / 100 ) = 91.0