未来十年计算机科学研究需要的四种思维

刘宇航
中国科学院计算技术研究所

关键词：历史思维　计算思维　数据思维　结构思维

引言

我国天河2号超级计算机已经连续六届居世界第一[1]，但是截至目前还没有中国国籍的科学工作者在计算机科学领域获得图灵奖。这表明，我国与美国的计算机研究水平相比，局部虽有突破，但整体上仍有差距。美国对向我国出口处理器芯片实行了限制，并提出限期10年率先研制成功E级超级计算机的计划[2]，为我国高性能计算的赶超设置了障碍。在这种国际竞争日益激烈的形势下，如何提高计算机科学研究水平，明确研究的道路和目的是值得我国计算机科学工作者深入思考的问题。

贝弗里奇(1908~2006)在1957年著的《科学研究的艺术》[3]中总结归纳了孟德尔(Mendel, 1822~1884)、费歇尔(Fischer, 1852~1919)等经典自然科学家的研究方法。当时计算机科学还很年轻，贝弗里奇不可能介绍或总结计算机科学研究方面的规律。今天，计算机发展的历史已经70年，我们有必要探析一下计算机科学研究了。计算机科学研究作为高级的思维活动，需要科学工作者具备历史思维(historical thinking)、计算思维(computational thinking)、数据思维(data thinking)、结构思维(architectural thinking)四种必要的思维能力。

历史思维

历史思维是从历史中找到定位、借鉴经验、发现机遇的古为今用的思维。长期以来，我国在中学教育阶段就开始了文理分科，历史被划入文科学习范畴，很多从事计算机专业的人员因为没有受到这方面的系统教育，不了解社会文明发展历史，不了解科技发展历史，缺乏历史知识和历史感悟能力。计算机研究人员进行的与培养历史思维相关的活动是阅读文献，进行相关工作的文献综述，但这项工作本身仅是狭义的知识性的调研，即使这样，很多人也没有做好。但是，历史思维不是“花瓶”式的点缀，而是进行科学创造必须具备的催化剂，具有以下重要作用。

首先，了解历史，通过历史思维，可以在人类文明发展史中找到自己的价值定位，培养科学研究的兴趣。对于把计算机科学研究作为自己的职业乃至毕生奋斗的事业的人来说，明确道路和目的是十分重要的。目的高于道路，道路服务于目的。科学研究的目的在于认识和改造世界，提高生产效率，改善生活，提升幸福感。但是，由于现代科技活动分工过细，计算机科学工作者未必能清晰地看到自己的科研成果直接给人们带来的幸福感。在医学上，一种药物可以拯救几百万人的生命；在数学上，一个定理可以解放人类的思想；在计算机科学上，计算技术是第三次工业革命的主要标志，一种体系结构和一种算法，可以使计算更快速和高效，可以满足人类通过应用程序定义的各种需求。这些认识是

兴趣的源泉，只有通过历史思维来建立。

其次，了解历史，通过历史思维，可以学习优秀科学家坚忍不拔的意志，培养科学研究的定力。科研的道路并不平坦，富有探索性、曲折性和不确定性，需要有长期从事科学事业的定力。我国首位获得诺贝尔生理学或医学奖的科学家屠呦呦从1972年以来长期研究治疗疟疾的青蒿素，经历多次失败，献出毕生的精力，但一直是“三无教授”（没有博士学位、没有留洋背景、没有院士头衔）。年届60的数学家张益唐证明了“相邻素数之距离有限”这一困扰数学界一个多世纪的难题，但直到完成定理证明前仍是一名讲师[4]。这两位科学家的人生经历、研究经历都是发人深省的，都体现了“幽芳不为春光发，只待秋风。只待秋风，香比余花分外浓”的境界[5]，体现了不急于求成、对科学事业孜孜以求的精神，对计算机科学的研究具有启示意义。这些认识是定力的源泉，只有通过历史思维来建立。

第三，了解历史，通过历史思维，可以站在巨人的肩膀上进行增量式创新，发现科学研究的机遇和契机。在科学研究中，一个很有趣的问题是个人英雄主义重要还是集体主义重要？屠呦呦、张益唐等科学家的个体素质毫无疑问起到极为关键的作用，但是也要看到人类集体努力的作用。屠呦呦的研究继承了1700年前我国东晋医学家葛洪(284~364)的成果。张益唐研究成功的基础，可追溯到20世纪80年代以北京大学丁石孙教授为代表的扎实的基础数学教育，以及国际上2005年戈德斯顿和鲍姆等人的工作[4]。上溯至1900年，如果希尔伯特(1862~1943)没有明确提出这一问题从而引起学术界的重视，戈德斯顿和鲍姆等人的工作也是不可能出现的。科学工作者从来就不是一个人在单打独斗，而是在一个跨越时间、超越国界的集体中发挥自己独特的作用。基于这一思路，伊利诺理工大学孙贤和教授继1967的埃姆道尔定律(Amdahl’s law)和1988年的高斯塔弗逊(Gustafson’s law)定律之后，在1990年提出孙—倪定律[6](Sun-Ni’s Law)。在沿用多年的平均存储访问时间(AMAT)基础上，于2014年提出并发存储访问时间模型(C-AMAT)[7]。2015年，在孙—倪定律和C-AMAT模型的基础上，又提出同时受限于容量和并发的性能模型(C^2-bound)，有效应对了现代存储系统中不断增厚的存储墙[8]。图1表达了这一进展，从首台程序存储式大型计算机、首片微处理器到首片多核处理器，从埃姆道尔定律、高斯塔弗逊定律到孙—倪定律，从AMAT到C-AMAT，等等，展示了在历史进程中进行的增量式创新的时间序列。通过历史思维，可以立足重大科学问题研究的最前沿，站在巨人的肩膀上创造性地解决问题。

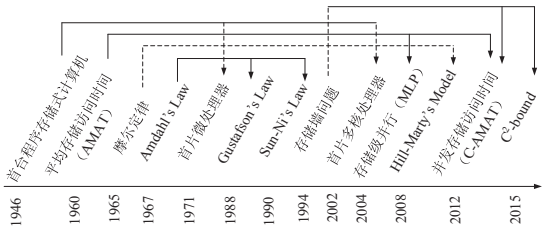

图1 与存储墙相关的计算机样机、定律与模型

具备了历史思维建立的兴趣和定力，以及机遇和基础，就可以从“计算”和“数据”出发，培养计算思维和数据思维。

计算思维

计算思维是尝试利用计算解决问题，将研究对象转化为可计算的问题的思维。计算机是机器，但不同于普通的机器。计算机可以运行无穷多的应用程序，具有无穷多的用途。最新获得诺贝尔化学奖的实验不是在传统的化学实验室中通过试管、试剂完成的，而是在超级计算机上通过模拟来揭示光合作用等化学过程[9]。计算已成为人类生产生活难以割舍的一部分，深刻影响着人类的生产、生活乃至思维方式。通过计算思维，人类运用数学知识对现实问题进行建模、求解，实现并行、预测、推理、聚类、抽象等功能。人类不再受限于感官接受的信息，不再受限于想象、联想、猜测，不再受限于心算，而是交给计算机去完成“计算”任务。人类从诞生至今的300万年的历史，是计算思维发展的历史。计算思维产生的时间要远远早于计算机诞生的

时间 (1946, EDVAC)，在计算机诞生之前人类已经发明了功能近似、机制迥异的多种计算工具，如算筹、算盘等（如图 2 所示）。

算筹　　算盘　　ENIAC　　天河2号

图2　古代计算装置与近现代计算机

计算思维的一个本质特征就是“求诸于外”，求诸计算机器，也就是周以真 (Jeannette M. Wing) 所说的“自动化”[17]。计算思维的发展过程，就是不断地从手工、半自动到全自动地“求诸于外”的过程，因此需要培养建模能力，使具体研究对象转化为可计算问题。冯·诺伊曼“程序存储”的计算机器体系结构（如图 3 所示），把指令序列作为广义的数据进行存储，然后按序执行，极大地提升了人类“求诸于外”的自动化程度，通过机器完成大量含有逻辑意义的计算操作，在很大程度上弥补了人类大脑运算不精确、不善于高速进行重复操作的缺陷。

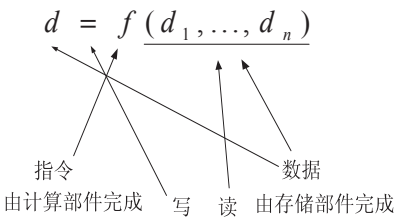

图3　冯·诺依曼“程序存储”式计算机的两大基本要素：计算与数据

数据思维

数据思维是以数据为中心，应用和设计计算机的思维。从大量数据中发现规律，是数据思维的体现之一。数据是计算的对象，也是计算的结果。信息存在于数据中，需要创新途径使现实世界中的模拟信号转化为可用字节表示的数据。数据思维研究的就是将研究对象转化为可用字节表示的数据。

数据思维的体现之二是以数据为中心设计计算机，提高系统的效率。大数据的高容量、低价值密度等特征对计算机体系结构提出了严峻挑战。冯·诺依曼结构以计算为中心，计算与存储是分离的，存储层级之间的延迟存在很大差异，芯片用于数据访问的管腿数量有限，这些带来了严重的延迟墙和带宽墙（有时两者统称为存储墙）问题，由此数据访问成为限制性能的瓶颈（如图 4 所示）。随着片上处理器核心数量的增加，存储墙越来越厚。深度多级高速缓存耗费了微处理器芯片 80% 以上的晶体管资源，例如，Power8 处理器采用一级数据缓存 64KB、一级指令缓存 32KB、二级缓存 4MB、三级缓存 96MB（所有核心共享）、片外四级缓存 128MB，相应地带来了能耗、温度的提升。但是高速缓存块在 59% 的时间里都是无用的 [10]。实现数据访问模式与底层存储系统之间的匹配成为亟待解决的问题 [11]。我国天河 2 号计算机虽然计算速度位居世界第一，但效率仅为 61.7%，同效率为 99.8% 的 IBM nx360M4 存在显著差距。仅有充分运用数据思维才能有效缩短这一差距。

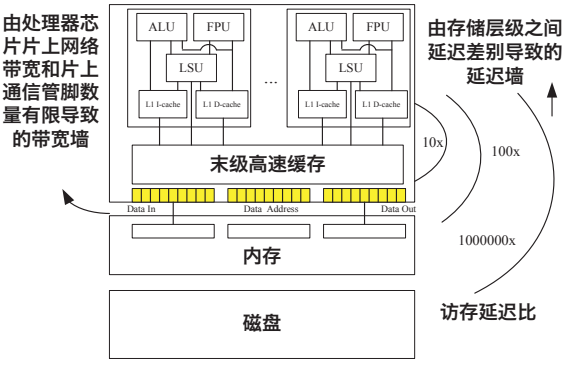

图4　数据访问成为计算机系统性能瓶颈

数据思维要求在设计计算机体系结构和算法时考虑原始数据的存储（取决于问题大小）、工作集的大小、存储访问的局部性和并发性，要对数据的采集、净化、存储、移动、运算等整个生命周期的全部阶段予以考虑。孙贤和教授在高性能计算领域较早地认识到存储问题是限制超级计算机性能的关键因素，在 1990 年提出了三大并行计算定律中唯一以数据为中心的孙一倪定律 [6]（如图 5 所示），在随后的 25 年始终以数据为中心聚焦研究存储墙问题，在

这个过程中一以贯之的就是数据思维。在大数据和超级计算并存且在一定程度上独立发展的今天，浮点运算能力不等同于数据处理能力，美国近期制定的未来 10 年计划已经注意到这一点，并着重强调 E 级超级计算机要处理 E 级字节数据 (10^{18}Bytes)[2]。未来高性能计算与大数据分析有可能相互融合，形成统一的生态系统 [9]。

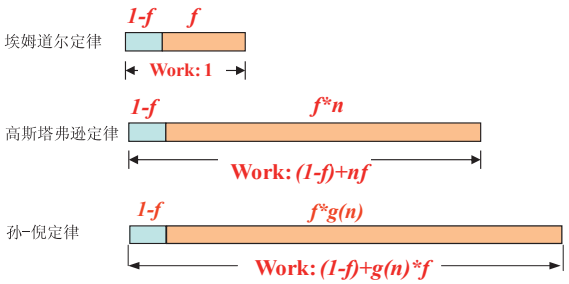

图5 孙-倪定律是并行计算三大定律中唯一以数据为中心的定律

结构思维

结构思维是通过创新系统的体系结构而不是单纯扩张系统的硬件资源的方法提升系统性能和效率的思维。据 2016 年 2 月的《自然》(*Nature*) 杂志报道，从 2016 年 3 月起，国际半导体技术路线图 (International Technology Roadmap for Semiconductors) 将停止对摩尔定律的追逐，延续了约 50 年的摩尔定律即将落幕 [17]。计算机的进步在后摩尔定律时代将更多地依靠体系结构创新。以超级计算机为例，2015 年 7 月，奥巴马签署总统令 [2]，要求美国在 2025 年之前实现百亿亿次 (exascale) 的计算速度。而现在世界上最先进的超级计算机是我国研制的连续五届蝉联冠军的天河 2 号，实际性能为 33 千万亿次 (petaflops)，仅实现百亿亿次的 3.3%。这就需要在 10 年内完成 30 倍的性能提升，平均每年要提高 40%。

这 40% 的性能提升从哪里获得和如何实现是值得研究的问题。根据摩尔定律 [12]，每 18 (36) 个月，晶体管的密度会翻一番，也就是每年增加 58%。按照波拉克法则 (Pollack's Rule)[14]，性能提升倍数为晶体管数量增加倍数的算术平方根，因此器件密度提高 58% 可以使性能提升 26% 。14% (28%) 的性能提升是不可能单纯从提升工艺的角度获得的。14% 和 28% 分别是在严格遵循或不严格遵循摩尔定律的假设下的估计。自 2010 年以来，芯片上器件密度提高的速度开始放缓，晶体管数量每 3 年翻一番 [13]，低于摩尔定律的预期。有研究指出，摩尔定律预计在 2020 年终结于 7nm 的工艺水平上 [15]，那时的晶体管的厚度将只有若干个原子，难以进一步缩小。

随着器件密度增速越来越背离摩尔定律，对计算体系结构创新的贡献率要求将越来越高(如图 6 所示)。

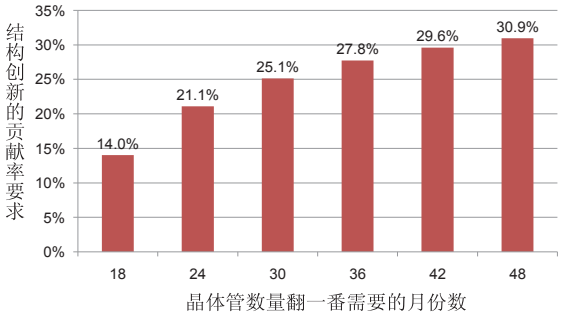

图6 结构创新的贡献率要求与摩尔定律满足程度之间的关系

导致摩尔定律放缓乃至失效的两个因素是：(1) 连接晶体管的导线越来越细，使得它们的电阻越来越大，无法承载足够的电流；(2) 单位面积上单位时间内消耗的能量越来越大，温度越来越高，导致可靠性越来越低。如果综合考虑片上器件密度未来几年增长预期，则每年计算性能提升的需求中 21% 要依靠计算机体系结构和算法的创新。

体系结构是算法的基础，决定了效率并最终决定效果。例如龙芯 3B1500 八核处理器，相对它的改进版龙芯 3B 在效率上有 35% 的提升，这主要归功于体系结构在两个方面的改善 [18]：一是存储层次结构的改善，包括末级缓存从 4MB 升级为 8MB，并为每个处理器核心引入 128KB 的 Victim Cache[1]；

[1] 从主高速缓存(main cache)中被替换的数据块称为牺牲者(Victim), Victim Cache称为牺牲者高速缓存, 用来缓存牺牲者。

二是对输入 / 输出结构的改善，包括超级传输协议 (Hypertransport) 从 1.0 升级到 2.0，内存接口从 DDRII 800 升级到 DDRIII 1200。

从实证的观点看，历史会给计算机设计者一些启示：1894 年 9 月中日甲午海战，我国北洋舰队在当时的亚洲实力最强，采用了雁形阵，目的是发挥舰首巨炮的优势；日本舰队因为没有财力装配巨炮，采用了纵形阵，在舰两侧装配大量的小口径炮，战役结果已为大家熟知。“少而巨”和“小而多”的结构成为战法的基础，是影响战争进程快慢和结果胜负的重要因素，这一点在计算技术的发展历程中也有所映射。从早期的巨型机到现在数万处理器核心的机群，背后是“少而巨”与“小而多”的权衡。以此类推，结构思维需要关注的维度有多核与众核、共享与私有、分布与集中、延迟与带宽、局部性与并发性、同构与异构、同步与异步、通用与专用等（如图 7 所示）。

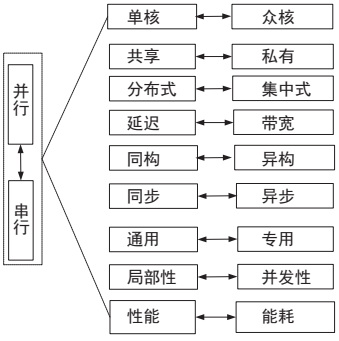

图7　结构思维的若干维度

这些维度很多，而且每个维度上都有多个选择，根据乘法原理，就构成了一个极其巨大的体系结构设计空间。例如有 10 个维度，每个维度有 10 种取值的可能，那么就存在 100 亿种 (10^{10}) 可能的结构。

与结构设计空间同时存在的还有一个庞大的应用负载空间。从理论上说，有无穷多种应用：从科学计算应用到事务处理应用；从计算密集型应用到数据密集型应用等。伴随着结构设计空间和应用负载空间的，有能耗、温度、面积等一系列约束。体系结构创新要在满足约束的前提下，在结构设计空间中“发现”一个匹配应用负载需求的最佳结构。由于结构设计空间和应用负载空间极其庞大，因此穷举所有的结构设计 - 应用负载的组合，确定对应的性能，然后发现最优的结构，是不可能实现的。只有依靠设计者的经验，综合利用推导分析、模拟等多种手段来完成这种选择。这正是体系结构创新的魅力所在。

在计算机科学中，计算机体系结构是算法的基础，决定着计算的效率，并最终决定服务质量。因此，我们要重视计算机体系结构和算法研究，通过创新结构，优化算法，提高效率，实现效果。

相互关系

历史是起点，计算是功能，数据是对象，结构是载体。历史思维为其他三种思维提供动机基础和工作基础。计算思维和数据思维要通过结构思维完成具体实现。

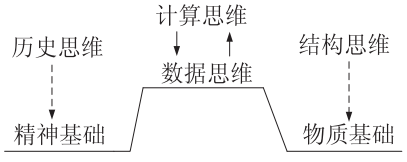

图8　历史和结构思维对计算和数据思维的支撑作用

从体系结构的角度看，计算操作部件和数据存储部件所构成的系统具有三个特点。(1) 过滤：计算部件可看成存储层次的第一级，高速缓存系统逐级起到过滤作用。(2) 并发：基于流水线、超标量、多线程、多核等并行技术，计算部件可同时发出多个数据请求，各级缓存也通过多体 (bank)、多端口 (port)、多通道 (channel) 等支持多个并发的数据访问。(3) 负反馈：当存储系统读写负荷较轻，可以快速提供数据时，计算部件可以继续发送更多的数据请求，当数据请求的数量增加到一定程度时，如果局部性差且存储系统的硬件并行度低于数据访问请求的并行度，将引起队列排队延迟和总线争用延迟，于是存储系统提供数据的能力降低，速度变慢，致使计算部件因为数据停顿而减慢了发送数据请求的速度，从而使存储系统的争用得以缓解，如此周而复始形成“负反馈环”。这一机制充分反映了数据与

体系结构的交织。从根本上说，可计算的前提是待处理信号可用字节表示，高速计算的前提是快速的数据移动。计算和数据之间是相互依赖的关系，计算思维不排斥以数据为中心的数据思维。

相关工作

2006年，时任美国卡内基梅隆大学计算机科学系主任、现任美国国家科学基金会(NSF)计算机和信息科学与工程部(CISE)主任的周以真教授，在《美国计算机学会通讯》(*Communications of the ACM*)上首次提出了计算思维的概念[16]：“计算思维是运用计算机科学的基础概念去求解问题、设计系统和理解人类的行为。它包括了涵盖计算机科学之广度的一系列思维活动。”而本文所述的“计算思维”特指“尝试通过计算解决问题，将研究对象转化为可计算的问题的思维”。本文同时讨论了历史思维、计算思维、数据思维、结构思维并论述了它们之间的关系，使之构成一个逻辑连贯的范式。

结束语

计算机诞生至今已70年，摩尔定律已生效50年，孙一倪定律已提出25年。未来十年，芯片器件密度的增长将越来越放缓而背离摩尔定律的预测，孙一倪定律所指出的数据访问对计算机系统性能的约束将越来越强，因此需要更多的体系结构创新，计算机科学技术才可能飞跃发展，并最终迎来大数据和E级超级计算机的时代。计算机科学工作者培养和研究历史思维、计算思维、数据思维、结构思维，对于提高我国计算机科学研究水平，做出世界一流科技成果具有重要作用。■

刘宇航

CCF专业会员。中国科学院计算技术研究所助理研究员。主要研究方向为高性能计算、计算机体系结构、并发存储系统。liuyuhang@ict.ac.cn

参考文献

[1] Supercomputing site. http://www.top500.org/.

[2] https://www.whitehouse.gov/blog/2015/07/29/advancing-us-leadership-high-performance-computing.

[3] Beveridge, W. I. B. (1951).The art of scientific investigation. New York: Vintage.

[4] Zhang, Yitang. Bounded Gaps Between Primes. *Annals of Mathematics*, 179.3 (2015):1121~1174.

[5] 李纲. 宋. 丑奴儿/采桑子.

[6] X. H. Sun and L. M. Ni. Another view on parallel speedup. in SC'90. IEEE Computer Society Press, 1990:324~333.

[7] Sun, X. H., & Wang, D. (2014). Concurrent average memory access time. *IEEE Computer*, (5),74~80.

[8] Yu-Hang Liu and Xian-He Sun, C2-bound: A Capacity and Concurrency Driven Analytical Model for Many-core Design, in SC'15. Texas, Austin, USA, Nov. 2015, 1~11.

[9] Daniel A. Reed, Jack Dongarra. Exascale Computing and Big Data. *Communications of the ACM*, Vol. 58 No. 7, 56~68

[10]S. Khan, Y. Tian, and D.A. Jimenez, Sampling Dead Block Prediction for Last-Level Caches, IEEE/ACM MICRO, Dec. 2010.

[11]Yu-Hang Liu and Xian-He Sun, LPM: Concurrency-driven Layered Performance Matching, in the 44th ICPP, Beijing, China, Sept. 2015, 1~10.

[12]Moore, Gordon E. (1965). Cramming more components onto integrated circuits. *Electronics Magazine*. p. 4.

[13]International Technology Roadmap for Semiconductors.

[14]S. Borkar, Thousand core chips: a technology perspective, in the 44th DAC. ACM, 2007, 746~749.

[15]Intel's former chief architect: Moore's law will be dead within a decade. http://www.extremetech.com/computing/165331-intels-former-chief-architect-moores-law-will-be-dead-within-a-decade.

[16]Jeannette M. Wing. Computational Thinking. *Communications of the ACM*, 2006, 49(3).

[17]M. Mitchell Waldrop. More than Moore. *Nature*, vol 530, February, 2016.

[18]W. Hu, Y. Zhang, L. Yang, et al. Godson-3B1500: A 32nm 1.35GHz 40W 172.8GFLOPS 8-core processor. IEEE ISSCC. 2013: 54~55.