

智能体自分裂数量最优解的计算方法

张可佳, 李春生[†], 胡亚楠, 富宇

(1. 东北石油大学 计算机与信息技术学院, 黑龙江 大庆 163318;
2. 黑龙江省石油大数据与智能分析重点实验室, 黑龙江 大庆 163318)

摘要: 提出一种提高超高频高维线性运算能力(UCP)的方法,描述了智能体模仿白细胞的分化抵御现象、计算自分裂最优数量的过程. 首先,通过计算时间缩减率筛选和归类影响UCP的重要因素,并以此构建标准宿主环境;其次,模拟微绒毛的逻辑结构设计环境探测器,并讨论不同条件下智能体自分裂最优数量的计算方法;然后,为准确描述宿主环境的线性运算能力,提出一种基于超越数精度计算的评价方法,并分析了不同测试函数对目标基函数的适用度;最后,通过计算准纳米晶体结构内弱互斥力的作用验证了所提出方法在运算时间、资源损耗等方面具有明显改善.

关键词: 智能体; 环境探测器; 自分裂; 超高频高维线性运算; 性能计算; 并行处理

中图分类号: TP18 **文献标志码:** A

Computing method about optimal spontaneous fission quantity of agent

ZHANG Ke-jia, LI Chun-sheng[†], HU Ya-nan, FU Yu

(1. College of Computer Information Technology, Northeast Petroleum University, Daqing 163318, China;
2. Heilongjiang Provincial Key Laboratory of Oil Big Data Intelligent Analysis, Daqing 163318, China)

Abstract: A method for improving the ultra-high frequency high-dimensional linear computing power (UCP) is explored. The process that agents simulates the differentiation to resist the virus and calculate the optimal quantity of spontaneous fission is described. Firstly, the time scalage is calculated to screen and classify the important indicators of UCP, and the standard host environment is established. Then, the environmental detector of logic structure design of microvillus is simulated, and the computing method for determining the optimal quantity of spontaneous fission in different conditions is discussed. Furthermore, to precisely describe the linear computing power of the host environment, an evaluation method based on transcendental number precision calculation is proposed, and the adaptability of different measure formulas for objective function is discussed. Finally, the weak exclusion force inside structure of the quasi-nanometer crystal is computed to verify the effectiveness of the proposed method.

Keywords: agent; environmental detector; spontaneous fission; UCP; performance computing; parallel processing

0 引言

力学系统仿真、矢量成像、微观晶体重构过程中经常涉及超高频高维线性运算(Ultra high frequency high-dimensional linear computing, UHC). 与病毒入侵类似,它具有高维性、高频率性和单一行为等特点,存在运算时间过长、资源消耗较高等问题,特别在泛集群环境下表现得尤为突出. 解决这一问题的方法通常包括算法优化和性能优化.

文献[1-5]讨论了通过改进全局优化算法提高循环函数的计算效率;文献[6]提出了一种深度评估方

法,提高了Laplace-Fourier波形计算的效率;文献[7]通过改进拉舍尔算法提高了伪Domz模的恒定解求解速度. 这些方法切实改变了工作机理,提高了计算速度,但算法针对性较强,可重复性和通用性不高.

智能体性能优化包括改进智能性和构建多智能体系统(MASs). 文献[8]采用GPU运算,改变宿主环境运算机制,提高了弹性波数值模拟的计算效率;文献[9]讨论了两种异构智能体在动力学分析中的效率,但是受到单一智能体的设计限制,单段函数的运算效率提升不高. MASs可以大幅提高运算效率,但

收稿日期: 2018-03-13; 修回日期: 2018-08-02.

基金项目: 国家自然科学基金项目(51774090, 21306022); 黑龙江省自然科学基金面上项目(F2015020); 黑龙江省教育规划重大项目(GJ2017006); 东北石油大学引导性创新项目(2017YDL-12); 东北石油大学国家基金培育基金项目(2017PYQZL-11).

责任编辑: 张化光.

[†]通讯作者. E-mail: csli_dmis@nepu.edu.cn.

是对智能体的控制与社交策略是十分困难的^[10-11],于是部分研究人员热衷于引入量子计算、二元分级等理论来提高单体运算能力^[12-15],虽然在一定程度上降低了社交难度,解决了超高频线性计算的高耗时间问题,但是设计结构过于复杂,可操作性不强.因此,如何在狭义环境中提出一种高效用的并发计算控制方法是研究人员急需解决的经典NP难问题之一.

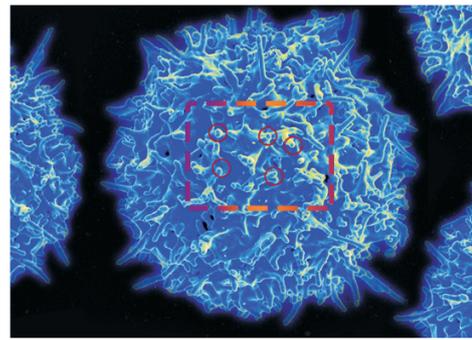
智能体是一种能够自行产生智能计算的行为个体,其行为激发过程^[16-18]与白细胞的模型结构和工作机理极其类似,换言之,细胞就是智能体的一种实例^[19]:在面对大量病毒入侵时,白细胞通过微绒毛感知自身能力和病毒感染强度,迅速分化形成一个简洁有序的“MASs”,并作出抵御行为^[20].本文模仿了这一过程,提出一种计算智能体最优分化数量(Optimal quantity of agent's spontaneous fission, OAS)的方法,指导智能体产生自分裂并开始并行计算,以达到资源消耗最少、运算效率最高的目的.

本文的主要研究成果和贡献如下:1)文献[21]通过计算加速度获取了DGMRES敏感特征,源于这一思想,本文通过计算时间缩减率获取影响UCP的重要因素,并以此构建一种评估UCP的标准宿主环境;2)阐述了从微绒毛到环境探测器的演化过程^[20],根据影响UCP的重要因素设计了一种环境探测器的抽象模型结构,并给出了环境探测器的逻辑组成部分;3)模拟了白细胞计算“入侵强度”和“生存能力”的过程,分析了在不同条件下“最优分化数量”的计算方法;4)提出一种基于超越数精度计算的宿主环境核心能力(The primary ability of host environment, TPA)评价方法,并给出了测试公式、适用范围及测试结果作为评价TPA的标准.

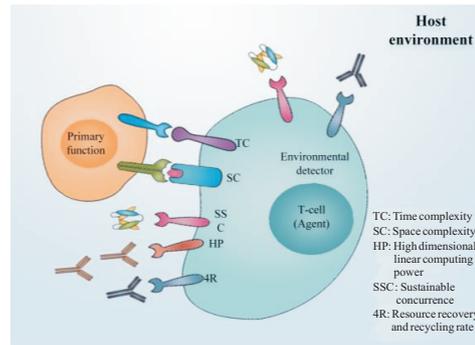
1 相关工作

类似于图1(b)所示的微绒毛探测范围,UCP涵盖了两类属性:宿主环境的计算能力和目标基函数的时空复杂度.宿主环境的计算能力包括TPA、次要能力(The secondary ability of host environment, TSA)和纤弱能力(Other supportive abilities of host environment, OSA);任务的时空复杂度包括目标基函数的时间复杂度和空间复杂度.本节分析了宿主环境和目标基函数的各项性能指标,获取影响UCP的重要因素并将其归类.在此之前给出宿主环境的真实场景定义.

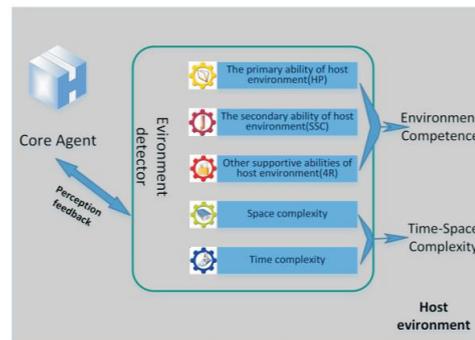
定义1 由VM构建环境参数完全相同的宿主环境副本 $HE_1HE_2 \cdots HE_n$,设置 HE_n 的环境参数集 Q_r 及其初始值作为初始标准环境.



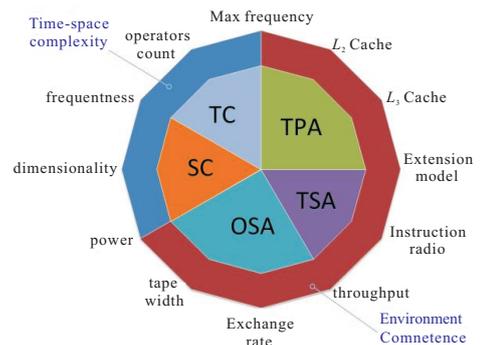
(a) 白细胞真实成像



(b) 白细胞感知结构



(c) 白环境探测器组成



(d) 探测内容及分层结构

图1 白细胞感知向智能体探测的演化过程

1.1 宿主环境计算能力的分析

根据CPU-z^[4-6]提供的物理性能指标,选取能够对宿主环境计算能力产生重大影响的环境参数集 Q_r .以测量到的PI浮点精度(106万位)的计算时间(42.6s)作为测量标准,分别测试18项物性参数的影响度.测试方法为:保持宿主环境的其他物性参数恒定,单项性能提高2~4倍,给出测试项目、测试值及测试时间如表1所示.

表1 计算能力的测试项目、测试值及测试时间

测试项目	初始值	测试值	t/s
CPU frequency	0.9 GHz	2.7 GHz	38.2
MAX frequency	1.3 GHz	3.9 GHz	7.8
Truck	2 GTs	6 GTs	41.8
L ₂ Cache	32 k	128 k	16.5
L ₃ Cache	2 Mb	6 Mb	11.7
SBS	7	9	41.9
Fre doubling	6	7	39.5
MAX FSB	1.33 k	2.66 k	40
Extension model	17	51	16.6
CAS delayed	8 t	24 t	-
Cycle time	18	54	44
tRFC	192	52	38.5
Instruction ratio	2T	4T	21.1
Throughput	30 Mb/s	90 Mb/s	18.4
Model capacity	512 p	1 024 p	39.7
Exchange rate	0.24	0.08	17
Tape width	416 M	1 024 M	20.8
DC Model	B	UB	42.6

时间缩减率的计算方法如下所示:

$$V_{ts} = \frac{t_{original} - t_{test}}{t_{original}} \quad (1)$$

依据各项性能指标的时间缩减率排序,并以相邻两项的缩减率差值大于2倍为标准^[6],即设定阈值 $V_{ts}^{VPT} > 0.45$,筛选结果包括 MAX frequency、L₂ Cache、L₃ Cache、Extension model、Instruction ratio、Throughput、Exchange rate、Tape width 等8项参数,并可给出如下结论:

- 1) TPA = MAXfrequency, L₃ Cache, L₂ Cache;
- 2) TSA = Extensionmodel, Instruction ratio;
- 3) OSA = Throughput, Exchange rate, Tape width;
- 4) 总时间缩减率 TPA > TSA > OAS.

综上,证明了三者重要程度依次减弱.

1.2 目标基函数的时空复杂度分析

时空复杂度包括时间复杂度和空间复杂度^[22]. 时间复杂度因素包括阶数 (pow)、维度 (dim)、频次 (fy) 和操作数 (oc). 操作数通常表现为目标基函数包含的运算符数量.

时间复杂度 O 定性地描述了某一计算过程的运行时间^[23]. 过程型目标任务的时间复杂度 $O_p =$ 单次任务执行时间 $t_o \times$ 执行次数 fy . 在线性计算中 $\forall t_o \propto b(\text{pow}, \text{dim}, \text{oc})$, 即在 h_0 下 $\forall \lambda \in \mathbf{R}$, 使 $t_0 = \lambda b(\text{pow}, \text{dim}, \text{oc})$, 因此时间复杂度可表示为

$$O_p(h_0, \varphi) = \lambda b(\text{pow}, \text{dim}, \text{oc})fy. \quad (2)$$

空间复杂度 S 表示目标基函数在运算过程中的资源占用度量^[24]. 除 pow, dim, oc 等参数外, 影响空间复杂度的弱势参数包括 Throughput (tr)、Exchange rate (er) 和 Tape width (tw). 与时间复杂度表示方式类似, 空间复杂度可表示为

$$S_p(h_0, \varphi) = \mu a(\text{tr}, \text{er}, \text{tw})b(\text{pow}, \text{dim}, \text{oc}). \quad (3)$$

2 OAS的计算方法

与多数细胞交流方式一样, 白细胞利用微绒毛的蛋白碎片探明 Leptospirose^[25] 的入侵强度, 通过附着接触感知宿主环境的抵御能力, 并根据生物电信号的刺激强度选择不同的反馈判断, 进而计算最优的分化数量. 蛋白碎片传递了能够描述病毒体积、分化速度、耐药性等重要特征的信息, 附着接触的机理是探明某些化合物在宿主环境母体液的浓度^[20].

环境探测器与微绒毛的感知机理相同. 图1给出了从白细胞感知向智能体探测的演化过程, 图1(a)是白细胞在高倍显微镜下的真实成像, 矩形圈定区域是微绒毛的密集区, 圆形区域描绘了微绒毛的微观形态; 图1(b)是细胞结构的直观示意图, 将 T-cell 细胞描述为智能体, TC、SC、TPA、TSA 和 OSA 分别表示环境探测器的5种感知器, TC 和 SC 描述了病毒的“入侵强度”, TPA、TSA 和 OSA 反应了宿主环境的“抵御能力”; 图1(c)诠释了图1(b)中的5种感知器, 并给出了环境探测器设计框架和组成结构; 图1(d)以雷达图的形式展示了图1(c)中环境探测器的具体探测内容, 同时描述了其详细分类和分层结构.

基于图1(d)所示的结构, 本节提出一种计算 OAS 的方法: 首先, 模仿白细胞计算病毒入侵强度的过程, 采用面积法分别表示宿主环境的计算能力和目标基函数的时空复杂度, 并计算入侵强度 (时空复杂度) 与抵御能力 (宿主环境的计算能力) 的差值; 然后, 借助图形法表达3种内部“生存能力”的强弱关系; 最后, 阐述不同条件下 OAS 的计算过程. 这种方法需要一个假设前提: 时空复杂度曲线是线性连续的^[23-24]. 方法的详细计算步骤如下.

Step 1: 定义一种 UCP 的表达框架如图2所示. Area[OCDR] 表示宿主环境的计算能力, 包括区域 Area[OCD] 和区域 Area[ODR]; 区域 Area[TOS] 表示

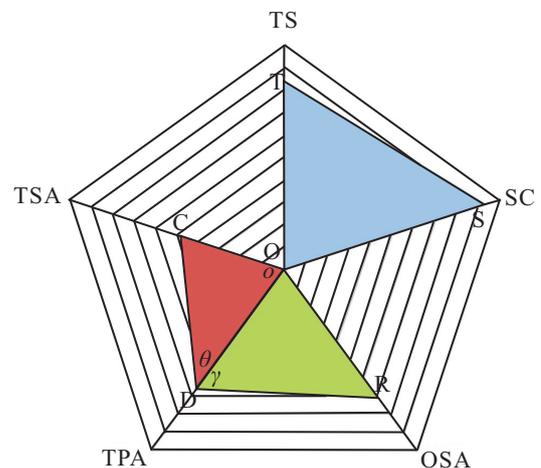


图2 UCP的雷达图表示框架

目标基函数的时空复杂度, o 为雷达图原点, 则面积差 w 的表达式为

$$w = S_{\text{OCDR}} - S_{\text{TOS}} \quad (4)$$

Step 2: 定义均衡阈值 $VPT(w) = 0, VPT(\theta) = VPT(\gamma) = \sin 0.3\pi$; 所有可能出现的条件组合为

$$Cd = \begin{cases} [Cd-1, Cd-6], & w > 0; \\ [Cd-7, Cd-12], & w < 0. \end{cases} \quad (5)$$

Step 3: 定义 $v_{\text{real}}^{\text{power}}$ 和 Ass_{power} 分别表示各项能力的物理值和分析值, $\phi(\langle w, \theta, \gamma \rangle)$ 表示最优解计算公式. 其中 Ass_{power} 表示各项能力的相对值, 有

$$Ass_{\text{power}} = \text{dimension} - \frac{v_{\text{real}}^{\text{power}} - v_{\text{standard}}^{\text{power}}}{v_{\text{real}}^{\text{power}}}, \quad v_{\text{real}}^{4R} \in [0, 1]. \quad (6)$$

Step 4: 根据表 2 得到智能体最优分裂数量的计算方法 $OAS = \text{Round}[\phi(\langle w, \theta, \gamma \rangle)], \min(OAS) = 1$.

表 2 不同条件下的 OAS 计算方法

w 值	编号	先验条件	计算方法	注释
	Cd-1	$\sin \gamma \geq VPT \geq \sin \theta$	不分裂	$\phi = 1$
	Cd-2	$\sin \gamma \geq \sin \theta \geq VPT$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{w}{\text{Area}[\text{OCDR}]} \langle \sin \theta, VPT \rangle v_{\text{real}}^{\text{SSC}}$	-
	Cd-3	$VPT \geq \sin \gamma \geq \sin \theta$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{w}{\text{Area}[\text{OCDR}]} \langle \sin \gamma, \sin \theta \rangle v_{\text{real}}^{\text{SSC}}$	-
$w > 0$	Cd-4	$VPT \geq \sin \theta \geq \sin \gamma$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{w}{\text{Area}[\text{OCDR}]} \langle \sin \theta, \sin \gamma \rangle v_{\text{real}}^{\text{SSC}} v_{\text{real}}^{4R}$	-
	Cd-5	$\sin \theta \geq VPT \geq \sin \gamma$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{w}{\text{Area}[\text{OCDR}]} \frac{Ass_{\text{HDP}}}{Ass_{\text{SSC}}} v_{\text{real}}^{\text{SSC}} v_{\text{real}}^{4R}$	弱势
	Cd-6	$\sin \theta \geq \sin \gamma \geq VPT$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{w}{\text{Area}[\text{OCDR}]} \langle \sin \gamma, VPT \rangle \frac{Ass_{\text{HDP}}}{Ass_{\text{SSC}}} v_{\text{real}}^{\text{SSC}} v_{\text{real}}^{4R}$	弱势
	Cd-7	$\sin \gamma \geq VPT \geq \sin \theta$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{\text{Area}[\text{OCDR}]}{\text{Area}[\text{TOS}]} \langle VPT, \sin \theta \rangle v_{\text{real}}^{\text{SSC}}$	-
	Cd-8	$\sin \gamma \geq \sin \theta \geq VPT$	$\phi(\langle w, \theta, \gamma \rangle) = v_{\text{real}}^{\text{SSC}}$	-
	Cd-9	$VPT \geq \sin \gamma \geq \sin \theta$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{\text{Area}[\text{OCDR}]}{\text{Area}[\text{TOS}]} \langle \sin \gamma, \sin \theta \rangle v_{\text{real}}^{\text{SSC}}$	-
$w < 0$	Cd-10	$VPT \geq \sin \theta \geq \sin \gamma$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{\text{Area}[\text{OCDR}]}{\text{Area}[\text{TOS}]} \langle \sin \theta, \sin \gamma \rangle v_{\text{real}}^{\text{SSC}}$	强势
	Cd-11	$\sin \theta \geq VPT \geq \sin \gamma$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{\text{Area}[\text{OCDR}]}{\text{Area}[\text{TOS}]} \frac{Ass_{\text{SSC}}}{Ass_{\text{HDP}}} v_{\text{real}}^{\text{SSC}}$	强势
	Cd-12	$\sin \theta \geq \sin \gamma \geq VPT$	$\phi(\langle w, \theta, \gamma \rangle) = \frac{\text{Area}[\text{OCDR}]}{\text{Area}[\text{TOS}]} \langle \sin \gamma, VPT \rangle v_{\text{real}}^{\text{SSC}}$	强势

3 TPA 的评价方法

TPA 的重要程度始终高于 UCP 的其他能力, 因此准确评价 TPA 是计算 OAS 的关键. 本节探讨了 TPA 的重要指标敏感程度, 并提出了一种基于超越数精度计算的 TPA 评价方法.

3.1 指标敏感程度分析

给出测试样本 PI 的浮点精度集合 $U = 0.42, 1.06, 3.37, 8.42, 20.11, 50.67, 130.44, 310.23$ (单位: 百万位); 设定测试值为标准宿主环境下初始值的 1.05 倍; 通过时间缩减率分析 TPA 各项指标的敏感程度变化

趋势, 测试结果如表 3 所示, 其中 up 表示提升倍率, uts 表示时间缩短率.

定义 $VTS(M - L_3)$ 、 $VTS(L_3 - L_2)$ 分别表示 MAX frequency 与 L_3 Cache、 L_3 Cache 与 L_2 Cache 的时间缩减率差值, 分析并给出可观察到的结果如下描述: 1) 3 项指标的敏感度会随着精度的提高而加速变化; 2) 小幅提高测试值对 L_2 Cache 和 L_3 Cache 的敏感度变化不大; 3) 当精度足够高时, MAX frequency 的作用远远超过其他两项; 4) MAX frequency, L_3 Cache, L_2 Cache 的敏感程度依次递减.

表 3 不同条件下的 OAS 计算方法

精度 (megabit)	标准时间	MAX frequency		L_3 Cache		L_2 Cache	
		1.05 up/s	VTS	1.05 up/s	VTS	1.05 up/s	VTS
$u_1 = 0.42$	15.2	15.02	0.012	15.08	0.008	15.12	0.005
$u_2 = 1.06$	42.6	41.15	0.034	41.92	0.016	42.22	0.009
$u_3 = 3.37$	118.4	109.64	0.074	115.32	0.026	116.86	0.013
$u_4 = 8.42$	304.6	261.65	0.141	293.03	0.038	298.51	0.02
$u_5 = 20.11$	908	705.52	0.223	854.43	0.059	882.58	0.028
$u_6 = 50.67$	3017.4	2048.81	0.321	2791.1	0.075	2896.7	0.04
$u_7 = 130.44$	10425.9	5828.08	0.441	9466.72	0.092	9842.05	0.056
$u_8 = 310.23$	32031.6	11179.03	0.651	28347.97	0.115	29405.01	0.082

3.2 基于超越数精度计算的TPA评价方法

评价TPA的方法通常是分析浮点数算子精度计算的所用时间^[26-27]. 为满足线性收敛和规避算子穷尽, 本文选用超越数作为算子. 超越数(刘维尔数)指不满足任何整系数多项式方程的实数^[28], 具有无限性和收敛性两项典型特点^[29]. 选用超越数 $c = \text{PI}$, 定义测量公式 $\Phi(a, n)$ 和线性收敛级数 \hat{a}_{n-1}^n , 其中 a 表示目标精度(位数), n 表示激发态的运算频次. 由

于收敛级数影响目标基函数运算效率^[30], 给出6种常见的不同收敛级数的测量公式, 包括 Gosper Equation, DG Equation, BBP Equation, Vieta Equation, Maclaurin Equation, Integral Equation.

测试公式的基本属性 $P_a = \{\text{pow}, \text{dim}, \text{oc}, \hat{a}_{n-1}^n, \max(a) | \hat{a}_{n-1}^n = \lim(n) - \lim(n-1)\}$, 其中 $\text{pow}, \text{dim}, \text{oc} \in T$, $\max(a)$ 表示有效量程的极大值. 表4给出了6种测试公式的基本属性值以及在不同测试条件下的计算时间, 其中 $\text{m} = \text{megabit}$, $\text{b} = \text{billibit}$.

表4 6种测试公式的基本属性以及在不同测试要求下的计算时间

项目	Gosper	DG	BBP	Vieta	Maclaurin	Integral
基础属性	pow	4n	16n	n	—	2
	dim	n	n	1	—	1
	oc	5	6	2	2	4n
	\hat{a}_{n-1}^n	8	14	0.5	1	1
	max(a)	175 m	4.044 b	infty	infty	infty
计算时间/s	0.42 m	12	11	12	11	11
	1.06 m	292	331	40	31	79
	130.44 m	462	571	74	61	151
	310.23 m	582	751	101	78	238
	410.57 m	659	831	164	108	322
	540.26 m	724	906	292	163	415
	690.12 m	753	982	464	214	481
	790.56 m	781	1 022	623	284	561
	900 m	788	1 050	791	352	661
	1.24 b	801	1 072	876	425	724
	1.35 b	812	1 105	918	503	803
1.6 b	821	1 124	958	584	896	

基础属性是目标基函数选择测试公式的先决条件, 计算时间表示不同测试公式在不同精度下的标准运算时间. 以6阶多项式曲线拟合并预测5个周期后的变化趋势, 给出如下事实描述: 1) Gosper、DG 呈现了高阶性 ($\text{pro} = 4n, 16n$) 和高维性 ($\text{dim} = n$), 并且存在极限精度, 因此 Gosper、DG 适用于高阶线性函数的弱超高频环境; 2) BBP 收敛能力弱, 适用于低严谨线性回归计算; 3) Maclaurin 的操作数复杂, 适合复杂混合运算环境; 4) Integral 的维度较高, 适用于高维低阶目标基函数; 5) 与其他测试公式不同, Vieta 是一种迭代递归的方法, 具有逻辑分析能力, 但是无法准确评估其阶数和维度, 尤其适合于分段连续函数; 6) DG 的线性收敛速度极快, 可适用性也最强, 但计算耗时最长, 所以优先级最低.

TPA 的计算过程可详细地描述为以下步骤.

Step 1: 已知标准环境参数集 $\text{Qr}_{\text{origin}} = \{\text{qr}_{\text{origin}}^i | i \leq 18\}$, 量纲标准为5.

Step 2: 侦测并获取真实宿主环境参数集, 标记为 $\text{Qr}_{\text{real}} = \{\text{qr}_{\text{real}}^i | i \leq 18\}$, 总性能提升率如下计算:

$$P_{\text{up}} = \frac{1}{18} \sum_{i=0}^{18} \text{Gap}(\text{qr}_{\text{origin}}^i, \text{qr}_{\text{test}}^i). \quad (7)$$

Step 3: 根据总性能提升率及参数集选择测量公式 $\Phi(a, n) = \text{compare}(P_a, P_a^\psi)$.

Step 4: 计算目标基函数的测试频次和标准时间, 分别记作 $\text{frequency}(\varphi), t_{\text{HE}n}$.

Step 5: 计算总操作数 $\text{sum}(\text{oc}) - \psi$, 预计真实计算时间 $t_{\text{predict}} = t_{\text{HE}n} / P_{\text{up}}$.

Step 6: 测量真实计算时间 t_r , 计算高维线性运算能力如下所示:

$$\text{Ass}_{\text{dp}} = \begin{cases} \max(\text{Ass}_{\text{dp}}), & t_r < t_{\text{predict}}; \\ 5 - \frac{t_r - t_{\text{predict}}}{t_{\text{predict}}}, & t_r \in [t_{\text{predict}}, 6 t_{\text{predict}}]; \\ 0, & t_r > 6 t_{\text{predict}}. \end{cases} \quad (8)$$

4 实验测试

QNC 是一种由同一类原子/分子经过层级堆叠形成的正多面体结构, 通常以正二十面体和正十四面体形式出现. QNC 的弱互斥力计算包括空间距离计算和空间标准误差计算, 属于典型的高维线性运算. 实验以正二十面体晶体结构为例, 描述了智能体在任务处理前计算最优分裂数量的过程, 并提供不同数量智能体的测试数据, 通过对比验证了方法在处理

时间、响应时间、资源耗损等方面得到明显提高。

4.1 模型结构

图3(a)和图3(b)给出了4层和5层QNC的晶体空间结构. 晶体由20个同心正四面体堆叠形成, 图3(d)重构了正四面体的空间模型, 图3(b)的三角形区域展示了正四面体的截面. 实验目标是计算高层QNC的分子间弱互斥力, 给出前期准备工作和相关定义.

1) 定义 n 表示 QNC 层级, 以原点为中心, 标记 20 个正四面体 $T_d = \{T_u^a, T_m^b, T_d^c\}$, $a \in [0, 5]$, $b \in [0, 10]$, $c \in [0, 5]$. r 和 l 分别表示某正四面体 T_d 内 n 层 r 行 l 列的分子. 设置正四面体的空间向量系数 $\vartheta = [\sqrt{5} + 1]/4$, 扩张基数 $a = 1.0514$.

2) 定义目标基函数为空间距离函数 φ 和空间标

准误函数 Φ , 计算层级 $n = 9$ 、分子数为 2533 时的目标基函数总操作数和执行频次为

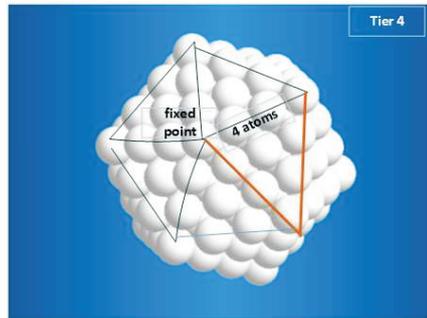
$$\text{frequency}(\varphi) = 0.5C_{\text{Atoms count}}^2 = 3.2 \text{ megabit};$$

$$\text{sum}(\text{oc}) - \varphi = \text{pow}(\varphi) \text{dim}(\varphi) \text{frequency}(\varphi) = 8.11 \text{ billibit};$$

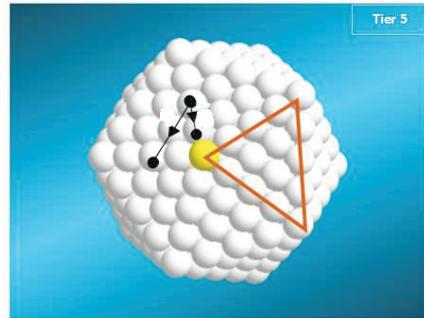
$$\text{frequency}(\Phi) = 0.5C_{\text{Atoms count}}^3 = 12.8 \text{ megabit};$$

$$\text{sum}(\text{oc}) - \Phi = \text{pow}(\Phi) \text{dim}(\Phi) \text{frequency}(\Phi) = 48.66 \text{ billibit}.$$

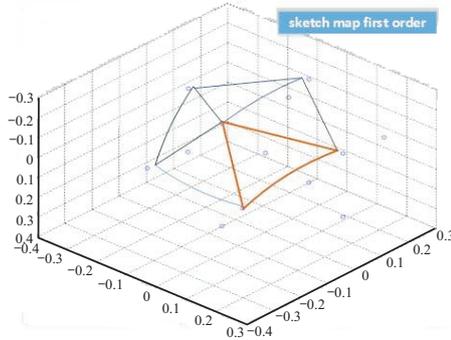
3) 实验宿主环境除以下参数外, 其他参数与标准宿主环境一致. CPU frequency = 2.7 GHz, MAX frequency = 5.4 GHz, Truck = 4 GTs, L_2 Cache = 128 k, L_3 Cache = 6 Mb, SBS = 15; Throughput = 512 Mb/s.



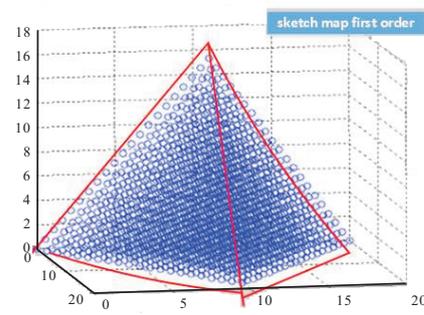
(a) 4层QNC晶体空间结构



(b) 5层QNC晶体空间结构



(c) QNC 晶体空间切面结构



(d) 正四面体空间模型

图3 QNC 晶体空间结构图

4.2 探测和计算过程

Step 1: 比较实验宿主环境 HEr 与 HEn 的各项物理参数, 计算总性能提升率 $P_{\text{up}} = 3.2$.

Step 2: 目标基函数的基础属性

$$\text{pow}(\varphi) = 2, \text{pow}(\Phi) = 3, \text{dim}(\varphi) = 2,$$

$$\text{dim}(\Phi) = 3, \hat{a}_{n-1}^n(\Phi) \rightarrow 0,$$

因此选用探测公式 Integral.

Step 3: 计算目标基函数的执行频次

$$\text{frequency}(\varphi) = 3.2 \text{ million},$$

$$\text{frequency}(\Phi) = 8.11 \text{ billion},$$

因此选用测试精度 $a = 105.4 \text{ million}$, 预计标准计算

时间 $t_{\text{HEn}} \approx 243 \text{ s}$.

Step 4: 计算目标基函数的总操作数 $\text{sum}(\text{oc}) - \varphi = 12.8 \text{ million}$, $\text{sum}(\text{oc}) - \Phi = 48.66 \text{ b}$; 在 $P_{\text{up}} = 3.2$ 时, 预计实验宿主环境的计算时间 $t_{\text{predict}} \approx 81 \text{ s}$.

Step 5: 测量任务持续时间 $t_r = 278 \text{ s}$, 计算 $\text{Ass}_{\text{TPA}} = \text{Ass}_{\text{dp}} = 2.56$.

Step 6: 计算目标基函数的时间复杂度 Ass_{TC} 和空间复杂度 Ass_{SC} , 即

$$\text{Ass}_{\text{TC}} = O_{\text{rate}} = 5 \frac{O_p(\text{HEr}, \text{Integral})}{O_p(\text{HEn}, \varphi)} = 4.2,$$

$$\text{Ass}_{\text{SC}} = S_{\text{rate}} = 5 \frac{S_p(\text{HEr}, \text{Integral})}{S_p(\text{HEn}, \varphi)} = 4.7.$$

Step 7: 计算固定态 Ass_{TSA} 和 Ass_{OSA} 的分析值, 即

$$v_{real}^{TSA} = \text{count(Core)SBS} \frac{\text{cycle time}}{\text{frequency doubling}} = 45,$$

$$Ass_{TSA} = 2.43,$$

$$Ass_{OSA} = 5 \frac{(1 - \text{exchange rate} \times \text{cycle time})}{\text{SBS}} = 3.56.$$

Step 8: 由 $\text{Area[OCDR]} - \text{Area[TOS]} = -4.19$, $\sin \theta > \text{VPT} > \sin \gamma$ 可知 $\phi(\langle w, \theta, \gamma \rangle)$ 的先验条件满足 C_{d-11} , 因此智能体最优自分裂数量

$$\text{OAS} = \text{Round} \left[\frac{\text{Area[OCDR]} \text{Ass}_{SSC}}{\text{Area[TOS]} \text{Ass}_{HDP}} v_{real}^{SSC} \right] = 33.$$

4.3 实验过程及结果分析

以线程模拟智能体的自分裂副本, 环境的极限并发数 $TSA = 45$, 由前期实验可知如下两种情况: 1) 当 $\text{OAS} \leq 14$ 时, 智能体单体运算量超载, 积压任务过多导致系统宕机; 2) 当 $14 < \text{OAS} \leq 28$ 时, 各项性能均处于低利用率 ($\text{rate} \leq 0.45$), 因此实验重点分析 $\text{OAS} \in [29, 45]$ 的情况. 设置资源剩余率的警戒阈值为 0.05, 当峰值运算强度的稳定阈值 $S_{\text{peak}}^{\text{con}} \in [67.5 - 72.5]$ 时, 警戒阈值 $S_{\text{peak}}^{\text{warn}} = 92.5$; 分别监测并记录真实运算时间 t_{real} 、累计总时间 t_{total} 、存储消耗量 C_{resource} 、资源剩余率 rate 、累计响应时间 t_{res} 、峰值运算强度 $S_{\text{Peak}} \in [0, 100]$ 等参数, 监测数据记录如表 5 所示. 当 $\text{OAS} = 33$ 时, 任务运算时间最短, 验证了方法的准确性.

表 5 $\text{OAS} \in [29, 45]$ 的实验监测数据

os	t_{real}/s	t_{total}/s	t_{res}/ms	rate	C_{resource}	S_{peak}/s
<14	-	-	-	>80	≈14.6 G	2.2~5.9
28	>48000	-	<60	>50	≈16.4 G	6.5~62.5
29	46992.5	1362782.5	60	38.5	16.727 G	64.8
30	45663.1	1369893	65	35.9	16.913 G	65.9
31	44357.5	1375082.5	69	33.3	17.125 G	66.6
32	43024.8	1376793.6	76	31.1	17.321 G	67.8
33	41721.6	1376812.8	82	28.7	17.512 G	68.5
34	41855	1423070	84	25.8	17.701 G	69.2
35	42001.5	1470052.5	824	22.3	17.912 G	71.1
36	42204.3	1519354.8	902	19.9	18.109 G	72.3
37	43508.9	1609829.3	986	17.2	18.314 G	73.9
38	44902	1706276	1012	15.2	18.521 G	77.6
39	46403.2	1809724.8	1108	12.8	18.716 G	82.8
40	48092	1923680	1216	9.9	18.920 G	87.2
41	60000	246000	2327	7.2	19.114 G	92.3
>42	-	-	-	-	-	-

5 结论

本文提出了一种有效提高超高频高维线性运算效率的方法. 方法本身并不局限于文中所提到的测试公式、参数和算法, 可以根据目标基函数的形态和宿

主环境的结构作出相应改变. 现对本文总结如下:

1) 智能体最优自分裂数量的计算过程是任务执行前的预判, 可以有效地屏蔽很多无法真实计算的影响因素, 但没有考虑任务执行过程中的宿主环境变化;

2) 构建并量化了一种标准的宿主环境, 作为一种框架, 其制定的标准是相对的;

3) 选取典型特征描述时空复杂度的方法为判断目标基函数的难易程度提供了一种有效依据;

4) 超越数的精度计算方法可以根据实际需要引入新的测试公式, 但相关的测试标准需要重新定义.

实验计算和对比表明了本文所提出的方法在运算时间、资源损耗等方面具有明显提高. 由于本文提出的方法可以根据宿主环境的真实能力最大化发挥其性能, 因此可以较好地解决有限资源下的最优化任务分配和调度问题, 并已经在高分子微观建模、地震数据处理等领域发挥了一定作用, 大大缓解了计算资源的需求压力. 未来将以本文研究成果为基础, 研究在泛集群环境下处理计算密集型任务的智能体最优数量计算方法, 并提出一种假设: 智能体可以自发地形成联盟以解决复杂科学计算任务.

参考文献 (References)

- [1] McMullen M S, Jameson A. The computational efficiency of non-linear frequency domain methods[J]. J of Computational Physics, 2006, 212(2): 637-661.
- [2] Zheng Y, Peng M. Track truncation method for improving the computational efficiency of 2D matrix MOC[J]. Annals of Nuclear Energy, 2017, 109: 684-691.
- [3] Zhou D, Lu Z, Guo T. Improvement of computational efficiency of circular function-based gas kinetic scheme by using jacobian-free newton-krylov method[J]. Computers and Fluids, 2017, 161: 121-135.
- [4] Golbabai A, Nikpour A. Computing a numerical solution of two dimensional non-linear Schrödinger equation on complexly shaped domains by RBF based differential quadrature method[J]. J of Computational Physics, 2016, 322: 586-602.
- [5] 卢少磊, 方浩. 一种改进的多agent分布式联盟形成算法[J]. 控制与决策, 2017, 32(4): 632-636. (Lu S L, Fang H. An improved distributed coalition formation algorithm in MAS[J]. Control and Decision, 2017, 32(4): 632-636.)
- [6] Zhang D, Zhang X, Yuan J, et al. A method for improving the computational efficiency of a Laplace-Fourier domain waveform inversion based on depth estimation[J]. J of Applied Geophysics, 2016, 124: 130-138.
- [7] Blund A, Husfeldt T, Lyckberg I. Computing the

- permanent modulo a prime power[J]. *Information Processing Letters*, 2017, 125: 20-25.
- [8] Packo P, Bielak T, Spencer A B, et al. Numerical simulations of elastic wave propagation using graphical processing units—Comparative study of high-performance computing capabilities[J]. *Computer Methods in Applied Mechanics and Engineering*, 2015, 290: 98-126.
- [9] Zhang Q, Hao Y, Yang Z, et al. Adaptive flocking of heterogeneous multi-agents systems with nonlinear dynamics[J]. *Neurocomputing*, 2016, 216(C): 72-77.
- [10] Zhu Y, Zheng Y, Guan Y. Quantized consensus of second-order multi-agent systems via impulsive control[J]. *Neurocomputing*, 2017, 270: 27-33.
- [11] Mali Xing, Feiqi Deng. Distributed event-triggered observer-based tracking control of leader-follower multi-agent systems[J]. *Neurocomputing*, 2017, 273: 650-658.
- [12] Ge X, Han Q L, Ding D, et al. A survey on recent advances in distributed sampled-data cooperative control of multi-agent systems[J]. *Neurocomputing*, 2017, 275: 1684-1701.
- [13] Yu M, Wang H, Xie G, et al. Event-triggered circle formation control for second-order-agent system[J]. *Neurocomputing*, 2017, 275: 462-469.
- [14] Terán J, Aguilar J, Cerrada M. Integration in industrial automation based on multi-agent systems using cultural algorithms for optimizing the coordination mechanisms[J]. *Computers in Industry*, 2017, 91: 11-23.
- [15] Gaosheng Zhang, Jiahu Qin, Weixing Zheng, et al. Fault-tolerant coordination control for second-order multi-agent systems with partial actuator effectiveness[J]. *Information Sciences*, 2017, 423: 115-127.
- [16] Gunasekera K, Zaslavsky A, Krishnaswamy S, et al. Building ubiquitous computing applications using the VERSAG adaptive agent framework[J]. *J of Systems and Software*, 2013, 86(2): 501-519.
- [17] Bergenti F, Iotti E, Monica S, et al. Agent-oriented model-driven development for JADE with the JADEL programming language[J]. *Computer Languages Systems and Structures*, 2017, 50: 142-158.
- [18] Wooldridge M, Jennings N R. Intelligence agents: Theory and practice[J]. *Knowledge Engineering Review*, 1994, 10(2): 115-152.
- [19] Yu J, Krishnan K K. A conceptual framework for agent-based agile manufacturing cells[J]. *Information Systems J*, 2004, 14(2): 93-109.
- [20] Cai E, Marchuk K, Beemiller P, et al. Visualizing dynamic microvillar search and stabilization during ligand detection by T cells[J]. *Science*, 2017, 356(6338): 1-10.
- [21] Toutounian F, Buzhabadi R. New methods for computing the Drazin-inverse solution of singular linear systems[J]. *Applied Mathematics and Computation*, 2017, 294: 343-352.
- [22] Esquivel F J, Alonso F J, Angulo J M. Multifractal complexity analysis in space-time based on the generalized dimensions derivatives[J]. *Spatial Statistics*, 2017, 22: 469-480.
- [23] Petrella R J, Karplus M. Electrostatic energies and forces computed without explicit interparticle interactions: A linear time complexity formulation[J]. *J of Computational Chemistry*, 2005, 26(8): 755-787.
- [24] Eli Ben-Sasson, Nicola Galesi. Space complexity of random formulae in resolution[J]. *Random Structures and Algorithms*, 2003, 23(1): 92-109.
- [25] Vlk C, Ty L, Wf L, et al. Leptospirosis in human: Biomarkers in host immune responses[J]. *Microbiological Research*, 2018, 207: 108-115.
- [26] 李浩亮, 杨任农, 李秋妮, 等. 基于分布式PID控制的任意阶线性多智能体系统的一致性[J]. *控制与决策*, 2017, 32(5): 899-905.
(Li H L, Yang R N, Li Q N, et al. Consensus of multi-agent systems with general linear dynamics based on distributed PID control[J]. *Control and Decision*, 2017, 32(5): 899-905.)
- [27] Stepinski T, Uhl T, Staszewski W, et al. Numerical simulation of elastic wave propagation[M]. *Advanced Structural Damage Detection: From Theory to Engineering Applications*. 2013: 17-56.
- [28] Mahler K. The successive minima in the geometry of numbers and the distinction between algebraic and transcendental numbers[J]. *J of Number Theory*, 1986, 22(2): 147-160.
- [29] Waldschmidt M. Auxiliary functions in transcendental number theory[J]. *Ramanujan J*, 2009, 20(3): 341-373.
- [30] Collier N, Dalcin L, Calo V M. On the computational efficiency of isogeometric methods for smooth elliptic problems using direct solvers[J]. *Int J for Numerical Methods in Engineering*, 2015, 100(8): 620-632.

作者简介

张可佳(1986—), 男, 讲师, 博士, 从事智能优化理论、多智能体系统等研究, E-mail: zkj@nepu.edu.cn;

李春生(1960—), 男, 教授, 博士生导师, 从事智能优化理论、数据挖掘等研究, E-mail: csli_dmis@nepu.edu.cn;

胡亚楠(1991—), 女, 博士生, 从事数据挖掘、多智能体系统的研究, E-mail: hyn@nepu.edu.cn;

富宇(1974—), 男, 副教授, 从事智能优化理论、油田大数据集成等研究, E-mail: dqfuyu@nepu.edu.cn.

(责任编辑: 孙艺红)