
通用晶体结构预测软件:
进化晶体学

A.R. Oganov, C.W. Glass, A.O. Lyakhov, Q. Zhu, G.-R. Qian, H.T. Stokes,
P. Bushlanov, Z. Allahyari, S. Lepeshkin

with contributions from

M. Davari, R. Agarwal, X. Dong, M.S. Rakin, P. Perterra, Z. Raza,
M.A. Salvado, D. Dong, Q. Zeng

用户手册

Version 9.4.4, June 21, 2017.

© A.R. Oganov, with sections by Q. Zhu, M.S. Rakin, G.-R. Qian and A. Samtsevich

中文统稿: 曾庆丰, 张瑾

翻译: 刘宁, 李端, 李坤, 程翔, 樊涛, 姚茜, 魏晓婷

<http://uspex.stonybrook.edu>



Oganov's Lab

COMPUTATIONAL MATERIALS DISCOVERY LABORATORY

Contents

1	USPEX的特点, 目标和历史	4
1.1	概述	4
1.2	USPEX的特色	8
1.3	有关USPEX主要原理及应用的文章	9
1.4	版本历史(只列举最重要的版本)	10
2	初始化	14
2.1	如何获得USPEX	14
2.2	版权引用	14
2.3	错误报告	14
2.4	运行USPEX的机型配置	14
2.5	与USPEX兼容使用的代码	14
2.6	如何安装USPEX	15
2.7	如何运行USPEX	16
2.8	运行USPEX示例: 迷你手册	17
2.8.1	测试USPEX Python运行器	18
2.8.2	运行例子EX13-3D_special_quasirandom_structure_TiCoO	18
2.8.3	运行一个与外部代码交互的示例	19
2.8.4	检查结果	20
3	输入与输出文件概述	21
3.1	输入文件	21
3.2	输出文件	21
3.3	Specific/ 文件夹	25
3.4	Specific/文件夹里的INCAR_*文件 (VASP)	26
4	输入文件INPUT.txt 的设置	29
4.1	运行的类型和系统	29
4.2	种群	35
4.3	适者生存和选择	36
4.4	结构产生和变异操作	37
4.5	约束条件	40

4.6	晶胞	42
4.7	重新开始	44
4.8	详述从头算法	44
4.9	指纹设置	47
4.10	反种子设置	48
4.11	空间群的判定	49
4.12	给开发者的关键词	50
4.13	不常使用的关键字	51
5	特殊情况的额外输入	55
5.1	分子晶体的MOL_1, MOL_2, ... 文件	55
5.1.1	分子晶体, calculationType=310/311	55
5.1.2	聚合物晶体, calculationType=110 (“线性链状模型”)	55
5.1.3	经典力场的额外输入	56
5.1.4	如何准备MOL文件	57
5.2	表面	57
5.3	变成分代码	60
5.4	进化准动力学命令	61
5.5	粒子群算法 (PSO) 代码	65
6	相变路径预测	67
6.1	变胞微动弹性带 (VCNEB) 法	67
6.2	VCNEB的Input选项	67
6.3	如何设置VC-NEB计算的初始路径	74
7	在线工具	76
7.1	结构表征	76
7.2	性质计算	76
7.3	分子晶体	76
7.4	表面	77
7.5	其它	77
8	常见问题	78
8.1	如何将结果可视化?	78

8.2	如何避免陷阱?	78
8.3	什么是单块计算?	78
8.4	如何使用种子技术?	79
8.5	如何巧妙考虑组成?	80
8.6	如何建立一个从本地机器到远程集群的密码连接?	81
8.7	如何从一个损坏了的*.mat文件重新开始计算?	82
8.8	当USPEX在一段时间内不运行, 该如何操作?	83
8.9	如何建立一个作业提交脚本?	83
8.9.1	第一步: 在提交文件夹Submission/中配置文件	83
8.9.2	第二步: 定期运行USPEX	86
8.9.3	Crontab	86
8.9.4	Shell脚本	87
8.10	如何使对称性代码在32位机器上工作?	87
9	附录	88
9.1	例子目录	88
9.2	测试运行	90
9.3	INPUT.txt输入文件示例	91
9.3.1	Fixed-composition USPEX calculation (calculationType=300):	91
9.3.2	Variable-composition USPEX calculation (calculationType=301):	92
9.3.3	Evolutionary metadynamics (calculationMethod=META):	93
9.3.4	vcNEB calculation (calculationMethod=VCNEB):	94
9.4	空间群列表	95
9.5	平面群列表	96
9.6	点群列表	97
9.7	USPEX中使用的共价原子半径表	98
9.8	USPEX使用的默认化合价表	99
9.9	USPEX使用的默认goodBonds表	100
	Bibliography	101

1 USPEX的特点，目标和历史

1.1 概述

USPEX是通用结构预测器的简称：进化晶体结构。在俄语里，“uspek”代表“成功”，鉴于此方法的高效，以及通过这种方法产生了许多有用结构，这一命名是恰当的！USPEX在计算材料发现方面具备许多独特功能，这里将列出它的一系列特点。

最初从2004年，USPEX项目的主要目的是不依赖经验知识预测晶体结构。这一目的已通过整合局部优化和实空间从头算法模拟，发展成为先进的进化算法。除了完全非经验式的搜索外，USPEX同时能够预测大量亚稳结构，并运用不同程度的先验知识进行不同类型的模拟。自2010年起，我们的代码迅速地涉及了其它许多类型的问题，从2012年起代码包含了许多互补的方法。

晶体结构预测是一个较早期的问题，实际上包括理论晶体化学的中心问题也很悠久。John Maddox¹于1988年写道

显然，目前的问题是全局优化，即在结构变动过程中发现晶体的最小自由能（每摩尔）。为了得到一些可能存在的结构，让我们考虑一种简化的情况，一个体积为 V ，有 N 个相同原子的固定立方晶胞。为了进一步简化，我们假设原子只能离散分布在一个分辨率为 δ 的网格节点上。这种数字离散化使得不同原子坐标组合的数目 C 有限化：

$$C = \frac{1}{(V/\delta^3)} \frac{(V/\delta^3)!}{[(V/\delta^3) - N]!N!} \quad (1)$$

如果将 δ 设为特征键长（例如， $\delta=1 \text{ \AA}$ ），由方程 1给出的组合数量将是自由能局部最小的结构数量的合理估计。如果有不止一种原子类型，不同结构的数量将显著增加。假设一个典型体积 $\sim 10 \text{ \AA}^3$ 的原子，考虑到斯特灵公式($n! \approx \sqrt{2\pi n}(n/e)^n$)，在一个有10原子的单胞内A元素（化合物AB）的可能结构数为 10^{11} (10^{14})，若在20个原子的单胞内为 10^{25} (10^{30})，30个原子的单胞内为 10^{39} (10^{47})我们可以看到，这些数目很大，甚至是总原子数 $N \sim 10$ 的小体系实际处理起来也不可能实现。更糟的是，随着 N 的增加其复杂性呈指数增加。很明显，除了最简单的含有 $\sim 1-5$ 个原子的单胞体系，在自由能表面上逐点探索所有结构也不可行。

USPEX^{2,3}采用的进化算法由A.R. Oganov 和C.W. Glass设计，还有随后做出很大贡献的A.O. Lyakhov 和Q. Zhu。它的高效来源于精心设计的变异运算，其可靠性很大程度上取决于在进化算法中运用了目前最先进的从头算法模拟。进化算法的最大优势是不需要任何体系的特殊知识（除了化学成分之外），同时还可以自我改进，即随后代的不断增加，越来越好的结构会被发现并用于产生新结构。下图(Fig. 1)是一个能量面（或性质）的可能区域放大图。而且，基于变异算法的灵活特性，很容易把一些附加的特点融入到进化算法中。

我们开发USPEX的一个主要动力就是2004年^{4,5}发现了 MgSiO_3 的后钙钛矿结构(Fig. 2)，这一发现显著改变了地球内核的结构模型。2010年9月，当USPEX公开发布时，用户数近200。用户增长速度很快——在2012年5月我们拥有800用户，这一数字在2014年12月已超过2100。

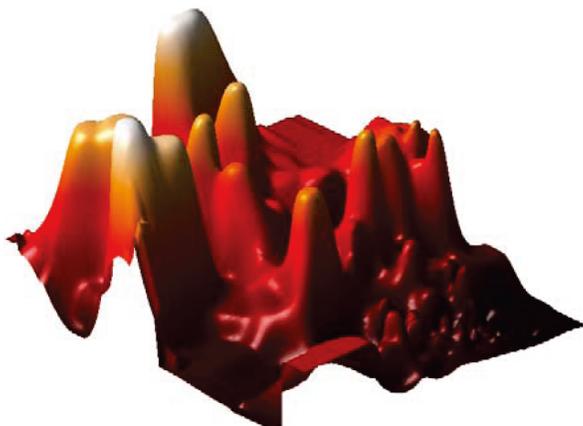


Figure 1: 图1: Au_8Pd_4 的简化面2D投影图, 显示了某一区域内具有低能量结构的聚集区。此图是由Oganov & Valle (2009) 的方法获得。

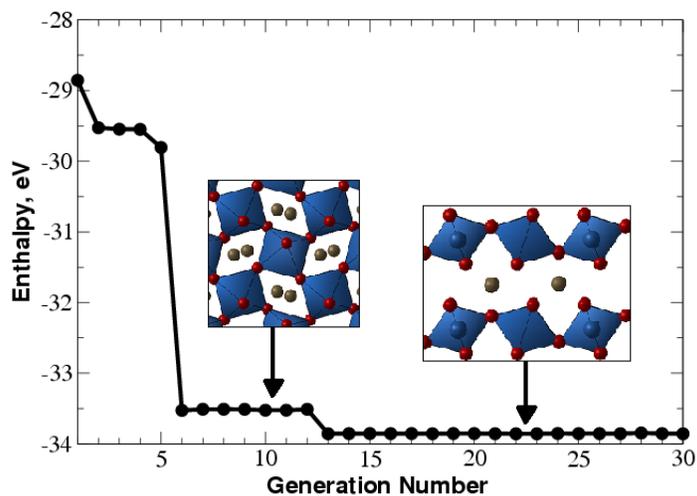


Figure 2: 在120 GPa下的 MgSiO_3 晶体结构预测 (20 原子/晶胞)。上图展示了随着代数的增加, 最优结构的焓值变化。在第六代和第十二代间最优结构为钙钛矿, 但在第十三代全局最低的 (后钙钛矿) 被发现。这次模拟计算在2005年结合了USPEX的第一个版本和从头算法完成。它没使用实验数据, 证明了USPEX可以在一次模拟中同时找到稳定结构和低能量的亚稳结构, 每一代都包含30个结构。这个图举例的是用最初的USPEX版本计算~10次中最慢的一个, 即使这样速度也相当快。

USPEX的普及归功于它的高效性和可靠性，而且在第一次对无机晶体结构预测的盲测⁶中体现出了USPEX优于其他方法测试，因为它是针对性的测试（模拟退火和随机抽样）。随机抽样（由Freeman和Schmidt分别于1993年和1996年提倡用于晶体结构预测，并于2006年被Pickard⁷以AIRSS的名字重新使用）是最简单的，也是成功率最低和计算成本最高的方法。由于在结构搜索中复杂的指数标度（公式1），USPEX的优势伴随体系大小的增加呈指数增长。然而，即使在小体系中，如有8个原子的GaAs单胞内，这些优势还是很大（在这个例子中，随机抽样需要平均500个结构弛豫来发现基态，而USPEX仅仅需要~ 30个结构弛豫就能找到（图3））。例如：由随机抽样预测得到的SnH₄的稳定结构⁷中三分之二都被证明不稳定⁸；同样，使用随机抽样⁹对N¹⁰和SnH₄的预测也被证实不正确（对USPEX¹¹和随机抽样¹²的预测结构进行比较）。

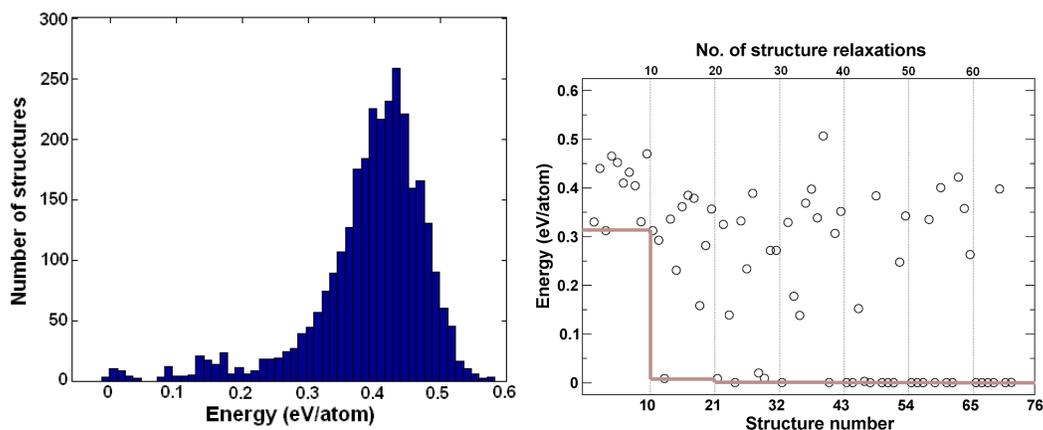


Figure 3: GaAs结构预测。a) 弛豫随机结构的能量分布图, b) 一个进化模拟进程图（细垂直线表示结构代数，而灰线则表现随代数变化最低能量的变化趋势。所有的能量都是基态结构能量的相对值，进化模拟中每代会有10个结构。此外，上一代具有最低能量的结构会延续到下一代）。

对于较大的体系，随机抽样倾向于产生几乎具有相同能量的完全无序结构，这样就成功率降低到几乎为零。从MgSiO₃超胞中有40个原子的后钙钛矿的例子可以看出：在120,000次的弛豫后随机抽样仍然没有找到正确的结构，而USPEX在几百次的弛豫后就找到了（Fig. 4）。

在USPEX中实现随机抽样是非常容易的，但大多数情况下只在测试的时候觉得这样很有用。同样，粒子群优化（PSO）算法和晶体结构预测(由A. I. Boldyrev开发，Wang, Lu, Zhu 和Ma改进实现) 已经被改造。它是在USPEX的基础上进行略微调整被作为修正的PSO (croPSO) 算法，它优于之前未修订版本的PSO，但实际上所有现存的PSO方法远不如USPEX那么高效可靠。如果有人想尝试，我们认为PSO方法更适用于测试。结合了Martoňák's的准动力学和Oganov-Glass的进化方法的进化准动力学方法¹³非常强大，弥补了进化算法的不足。对于全局优化和获得低能量的亚稳结构，这个方法是足够的，甚至可用于寻找可能的相转变路径。为了研究对于相变机制的详细信息，其他方法：变胞NEB法¹⁴ 和转变路径方法¹⁵ 加入到该版本¹⁶。

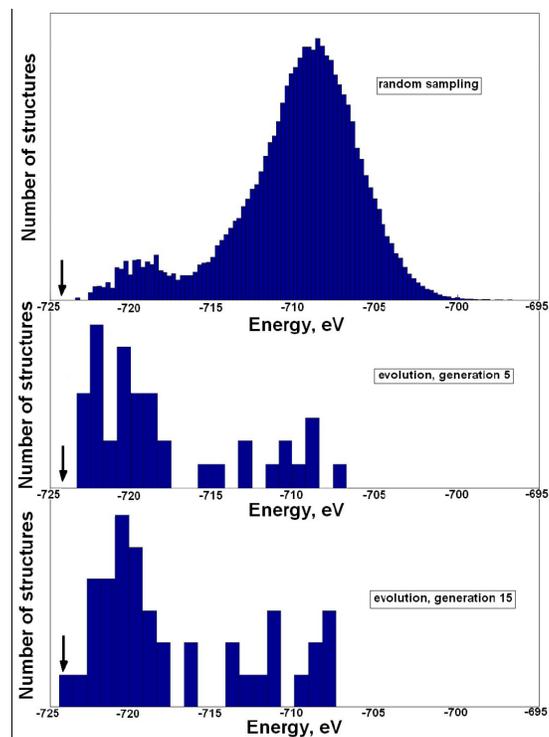


Figure 4: 能量表面取样：分别采用随机抽样和USPEX对有40个原子的 MgSiO_3 后钙钛矿相晶格常数进行计算并对比。局域优化结构的能量如图所示。随机抽样共生成 1.2×10^5 个结构（没有一种结构与基态一致）。USPEX的搜索，每一代包括40个结构，在15代内发现基态结构。基态结构的能量为箭头标示处。这张图显示，包含在进化搜索中的“学习”驱动模拟发现低能量结构。

1.2 USPEX的特色

- 只根据化学组成就能预测其稳定和亚稳定结构。还能根据元素模拟搜索稳定的组成和结构。
- 可以结合结构部分已知信息:
 - 可以根据实验获得的晶胞参数, 或者固定的晶胞形状, 又或者是固定的晶胞体积来对计算施加束缚条件(详见 4.6)
 - 可以从已知或者猜测的结构开始搜索; 算(详见 8.4);
 - 可以从预先确定的分子(包括结构灵活的分子)进行晶体结构组装(详见 ??)。
- 高效的束缚技术。可以消除在搜索空间累的非物理和无用的区域。晶胞缩小技术(Oganov & Glass, 2008)。
- 采用指纹函数的小生境技术。(Oganov & Valle, 2009; Lyakhov, Oganov, Valle, 2010)。详见 4.9.
- 采用随机的方式, 或者使用空间群和晶胞分裂技术来进行初始化(Lyakhov, Oganov, Valle, 2010)。
- 及时的结果分析— 空间群的确定(CIF形式的输出) (详见 4.11), 硬度的计算, 有序参量等, *etc.*
- 对纳米微粒结构和表面重构的结构预测。详见 5.2。
- 重启设置。使得计算可以从任何终止的时刻从新恢复并接着计算(详见 4.7)。
- 镶嵌在STM4代码里强大的可视和分析技术 (M.Valle) 可以完全地与USPEX进行接口(详见 8.1)。
- USPEX可以与VASP, SIESTA, GULP, LAMMPS, DMACRYS, CP2K, Quantum Espresso, FHI-aims, ATK, CASTEP, Tinker, MOPAC 等代码进行接口。详见 ??。
- 可以从本地的工作站向远程集群和超算提交作业。详见 8.9。
- 可以选择使用USPEX算法(默认算法), 随机取样法, 进化准动力学法(详见 5.5), 修正粒子群优化法(详见 5.4), 类最小跳跃阈来进行结构预测。可以使用进化准动力学法, 变胞NEB法(6.1)预测相变路径。
- 除了能量还可以选择以其他物理性质作为优化对象——例如, 硬度(Lyakhov & Oganov, 2011), 密度(Zhu et al., 2011), 带隙和介电常数(Zeng et al., 2014)和许多其它的性质。

- 为了方便开发者和用户，USPEX是一个基于MATLAB的代码同时它也能在Octave（一个类似于MATLAB环境的免费软件）——用户不需要进行任何编译，为了加强MATLAB版本的兼容性，只使用了基本的MATLAB命令。USPEX代码已经在MATLAB2012 版至2015版和Octave 3.4（新的 *Octave* 版本将不被支持）进行了开发和测试。
- 从9.4.1版本开始，USPEX具有了安装程序（`install.sh`文件）和一个基于Python的MATLAB代码（USPEX的Python模式），可以提供许多有用的命令行选项。

1.3 有关USPEX主要原理及应用的文章

1. Oganov A.R., Glass C.W. (2006). Crystal structure prediction using evolutionary algorithms: principles and applications. *J. Chem. Phys.*, **124**, 244704.
2. Oganov A.R., Stokes H., Valle M. (2011). How evolutionary crystal structure prediction works — and why. *Acc. Chem. Res.*, **44**, 227–237.
3. Lyakhov A.O., Oganov A.R., Stokes H., Zhu Q. (2013). New developments in evolutionary structure prediction algorithm USPEX. *Comp. Phys. Comm.*, **184**, 1172–1182.
4. Zhu Q., Oganov A.R., Glass C.W., Stokes H. (2012). Constrained evolutionary algorithm for structure prediction of molecular crystals: methodology and applications. *Acta Cryst. B*, **68**, 215–226.
5. Zhu Q., Li L., Oganov A.R., Allen P.B. (2013). Evolutionary method for predicting surface reconstructions with variable stoichiometry. *Phys. Rev. B*, **87**, 195317.
6. Zhu, Q., Sharma V., Oganov A.R., Ramprasad R. (2014). Predicting polymeric crystal structures by evolutionary algorithms. *J. Chem. Phys.*, **141**, 154102.
7. Zhou X.-F., Dong X., Oganov A.R., Zhu Q., Tian Y., Wang H.-T. (2014). Semimetallic Two-Dimensional Boron Allotrope with Massless Dirac Fermions. *Phys. Rev. Lett.*, **112**, 085502.
8. Lyakhov A.O., Oganov A.R., Valle M. (2010). Crystal structure prediction using evolutionary approach. In: Modern methods of crystal structure prediction (ed: A.R. Oganov), Berlin: Wiley-VCH.
9. Oganov A.R., Ma Y., Lyakhov A.O., Valle M., Gatti C. (2010). Evolutionary crystal structure prediction as a method for the discovery of minerals and materials. *Rev. Mineral. Geochem.*, **71**, 271–298.
10. Duan, D., Liu, Y., Tian, F., Li, D., Huang, X., Zhao, Z., Yu, H., Liu, B., Tian, W., Cui, T. (2014). Pressure-induced metallization of dense $(\text{H}_2\text{S})_2\text{H}_2$ with high- T_c superconductivity. *Sci. Rep.*, **4**.

1.4 版本历史(只列举最重要的版本)

- v.1 — 无局部优化进化算法。实空间表示，与VASP接口。实验版。2004.10。
- v.2 — CMA-ES的启用（CMA-ES是一个由N. Hansen设计的功能强大的全局优化方法）实验版。2005.1。
- v.3 — 有局部优化的进化算法。
 - v.3.1 — 工作版本，序列。主要的基本改进。
 - 3.1.4-3.1.5 — 第一产品版本。主要是片段移位和最小的亲代贡献的遗传（硬编码为0.25）。2005.5。
 - 3.1.8 — 适应k点网格。15/10/2005。
 - 3.1.11 — 从任意的一代开始重新计算。实验版。04 /11/2005。
 - 3.1.12 — 基于v.3.1.00的产品版本，变切片突变。11/11/2005。
 - 3.1.13 — 自适应缩放体积。29/1 /2005。
 - 3.1.14 — 初步的种子技术。29/11/2005（6/12/2005调试）。
 - v.3.2 — 大规模并行版本。
- v.4 — 统一并行/串行版本。
 - 4.1.1 — 晶格突变。20/12/2005（10/01/2006调试）。
 - 4.2.1 — 与SIESTA接口。初始群的大小可以不同于模拟运行中群的大小。24 /01 /2006（20/04/ 2006调试）。
 - 4.2.3 — 最佳结构的弛豫可选择。完全调试了的大规模并行版本。25 /04 /2006。
 - 4.4.1 — 与GULP接口。08/05/2006
- v.5 — 完全重写和调试版本，代码模块化。
 - 5.1.1 — Blanc, Gonzales。暂时放弃序列模式。特定原子突变，代码的互通性，速读INPUT_EA.txt中的参数。20/12/2006。
 - 5.2.1 — Z矩阵的SIESTA与接口，旋转变异运算，只确保远程作业提交（实验版）。01/03/2007。
- v.6 — 产品版，与本地和远程作业都可提交。
 - 6.1.3 — 可以高效地完成大体系的硬约束，USPEX中可实现优化。07/06/2007。
 - 6.2 — 开发版。
 - 6.3.1-6.3.2 — 引入单元格对角线的角度约束。完全重写的远程提交。改进输入格式。进一步扩展标准测试。07 /12 /2007。
 - 6.3.3 — X-com 网络界面（有S. Tikhonov 和S. Sobolev的共同参与）。在完善中。05/03/2008。
 - 6.4.1 — 指纹功能。07/04/2008。
 - 6.4.4 — 空间群识别（调试过，但仍有问题），可以选择性的关掉。快速指纹（参考表）。05/05/2008。
 - 6.5.1 — 针对大体系的分裂晶胞法。更容易的远程提交作业。可变数量的最佳结构（能量群）。16/07/2008。
 - 6.6.1 — 非常稳健的版本，与当地执行重新启用，改进了指纹和晶胞分裂技术。13/08/2008。

- 6.6.3 — 实现多种亲代遗传。01/10/2008。
- 6.6.4 — 增加亲代参与遗传的阈值。03/10/2008。
- 6.6.6 — 第一次实现多元指纹功能。04/12/2008。
- 6.6.7, 6.7.1 和 6.7.2 — 实现准熵测量种群的多样性，将CEL和SPF放入独立文件夹。10/12/2008。
- v.7** — 产品版本，含有变成分。
- 7.1.1–7.1.7 — Series of improved versions. Version 7.1.7 has been distributed to ~200 users. Variable composition partly coded, most known bugs fixed, improved tricks based on energy landscapes. Improved cell splitting, implemented pseudo-subcells. Implemented multicomponent fingerprints (much more sensitive to the structure than one-component fingerprints). 28/04/2009 (version finalized 28/05/2009).
- 7.2.5 — First fully functional version of the variable-composition method. Introduced transmutation operator and compositional entropy. 06/09/2009.
- 7.2.7 — 彻底调试，改进的重启功能，改进种子技术，与M. Baskes的MD++代码接口，引入了结构弛豫中的扰动和变成分的偏置适应度函数。25/09/2009，在7.2.8/9版本中进一步改进。
- 7.3.0 — 指纹完全支持变成分的代码，包括niching。“公平”算法用于产生组分的第一代结构。22/10/2009。
- 7.4.1 — 引入基于局部有序【22】坐标突变，遗传和原子置换也是基于局部有序。引入了硬度计算和基于硬度和密度的新型优化。04/01/2010。
- 7.4.2 — 调试，实现基于局部有序多亲代遗传。15/01/2010。
- 7.4.3 — 调试，实现新类型优化（最大化结构有序和种群多样性）。加入反种子技术，消除参数volTimeConst, volBestHowMany。24/01/2010。
- v.8** — 将新型优化写入产品版本。
- 8.1.1–8.2.8 — 开发版本。局部有序与坐标变异算子，软模变异运算，硬度的计算与优化，优化介质的敏感性。预测纳米粒子与表面结构，实现点群，软件整体表现提升。可选用PSO模拟，（由于PSO的低效率，不建议在实际应用中使用PSO模拟——所以只作测试用途）。参数goodBonds转变为矩阵用以建立纳米粒子。22/09/2010。
- 8.3.1 — 调试PSO算法，优化介电常数，清除输入。08/10/2010。
- 8.3.2 — 为集群引入连通性检查（非常有用），改进dynamicalBestHM=2选项，以及用于产生一代纯软模变异结构。改进集群的指纹功能。与Quantum Espresso 和CP2K接口。11/?10 / 2010。
- 8.4 — 开发版集合版，包含改进的反种子功能和纳米团簇计算上做了一些改进，建立表面重构、伪准动态和分子晶体分支程序。
- 8.5.0 — 第一代随机结构的初始化是使用H. Stokes添加的空间群代码。准动力学新的表达方式添加到一个单独的代码中。改正了varcomp, antiseeds, nanoparticles和计算硬度代码中的一些错误。18/03/2011。
- 8.5.1 — 工作界面的大量调试。对定晶胞，变成分和亚晶胞实现空间群初始化。20 /04 /2011。
- 8.6.0 — H. Stokes的工作为软件添加空间群判定程序。为分子晶体合并更新后代码（包括空间群初始化）。修正了一个SIESTA中错误（感谢D. Skachkov）。06 /05 /2011。

8.6.1–8.7.2 — 开发版，相当稳健。改进了定晶胞内的对称初始化。优化介电常数（使用GULP和VASP），带隙（使用VASP）和费米能级的DOS图（VASP）。启用图形输出。改进的软模变异运算（使用更好的标准模式与定向简并）和遗传运算（对实验物质使用能量规则相关性系数和余弦公式）。现在大多数变量有默认值，它减少用户输入量。减短并改进格式日志。13?/ 11 / 2011。

8.7.5 — 修正变成分的代码中错误，图形输出许多额外数据，如用于可变成分计算的原子近似体积。增加将结构提取为凸包图的功能，以便进行后期处理。21 /03 /2012。

v.9 — 产品版，增加客户体验，加入新类型的功能，在该领域设置新标准。

9.0.0 — 将进化准动力学和vc-NEB代码添加到USPEX软件包，增加准动力学的张量的版本，添加额外的数据和后处理工具，清理代码的输出。去除INPUT中的部分参数。改进软模变异。2012年4月。

9.1.0 — 发行版。清理并整合。用户人数超过800人。发布于28 / 05 /?2012。

9.2.0 — 运行GEM（广义进化准动力学）。持续开发GEM代码。空间群测定限度作为新的输入参数。改进大量置换变异参数。2012.7-2012.8。

9.2.1–9.2.3 — 改进GEM，种群多样性和超胞尺寸，改进选择模式。2012.9-2012.10。

9.2.4–9.2.6 — (9.2.4发行版)。多数输入参数采用智能默认值。改进的对称集群初始化能力。为纳米粒子增强有序性。新的参数用以调整空间群测定的限度。修正几个小错误。可以优化新的性质（准熵）。完全集成的vc-NEB的代码。2012.11-2012.12。

9.2.7. — 发行版。在没有结构弛豫的情况下进行合金的有序优化，（在不依托于特定的准随机机制而是通常用的普通随机机制的情况下，更易于产生准随机结构）。改进对称代（这点对于定晶胞计算非常重要）。在固定晶胞计算过程中，可以指定一个晶胞参数，不一定非是一个 3×3 矩阵，也可以是一排六个值（三个以埃为单位的长度值和三个以度为单位的角度值）。为了置换数目的最大化（参数howManySwaps），引入了智能默认值。添加新的测试，清除当前参数并启用旧参数。与CASTEP接口（感谢Z. Raza, X. Dong 和AL）。用户人数达到1160人。30 / 12 / 2012

9.3.0–9.3.3 — 修正在一个在随机对称结构代中出现的错误（这个错误出现于9.2.7版本）。显著简化输入和输出。创建OUTPUT.txt用以存放重要信息。对分子晶体进行晶体分裂。通过允许用户设定初始成分来改进变成分计算与LAMMPS和CASTEP的接口。添加新的测试案例。20?/ 03?/ 2013。

9.3.4 — 发行版，并做了清理。25/ 03/ 2013。

9.3.5 — 添加并调试了预测二维晶体的代码。19?/ 04?/ 2013。

9.3.6 — 集成和调试了GEM。改进变成分计算。二维晶体包含平面种群。29/ 04/ 2013。

9.3.8 — 一维聚合物晶体中添加平面群描述，改进变化学计量比的表面结构预测。19/ 06/ 2013。

9.3.9 — 发行版。显著改进的版本，加强用户体验，新功能（如二维晶体，GEM）表现良好，改进变成分算法（同时支持单块计算，即定组分但改变单胞内原子个数）、全功能表面计算，新类型优化（可优化介电常数、键长和新的介电材料的品质因数）。筹备在新的测试中与LAMMPS和ATK 接口。通过微小调试不断更新（最后调试于10 / 02/ 2014）。

9.4.1 — 主要升级，极大地提高用户使用性便利性（自动预估体积和每种案例的变异运算操作的比例），新功能（弹性性质的优化和Chen氏硬度模型，预测聚合物的结构、反组分、自动统计分析和改进的种子技术），第一次发布GEM，提供一系列USPEX的计算实例，测试案例，说

明文件。用户人数超过2100人。发布于30/12/2014。

9.4.2 — 发行版。兼容Octave 3.4中的错误。重新写入凸包图代码。与MOPAC接口。启用默认参数`goodBonds`, `valences`, `IonDistances`。三元、四元和更复杂成分的变成分搜索表现良好。用户超过2200人。发行于21 / 03 / 2015。

9.4.3 — 发行版。包括一系列的代码优化（这些更新将会加快代码的运行速度），与MOPAC的接口，改进了文件说明。这个版本几乎没有bug出现，此版本将会在版本10.中被进一步提高。发行于10/08/2015。

9.4.4 — 发行版。包括修复了空间群判断和用户反映的其它问题，改进了文件和例子。并可以完全兼容Octave 3.4和支持部分Octave 3.6/3.8/4.0。这个版本是近乎完全没有任何bug，具有里程碑式的意义，这个版本也会在版本10中包含。发行于05/10/2015。

2 初始化

2.1 如何获得USPEX

USPEX 是一款开源软件，可以从以下网址下载：

<http://uspex.stonybrook.edu>

在下载页面，分别有USPEX源代码包，示例和手册文件。

2.2 版权引用

无论什么时候使用USPEX，在所有出版物或者报告中必须在原文中指明，例如以下面这种方式：

“Crystal structure prediction was performed using the USPEX code^{2;13;17}, based on an evolutionary algorithm developed by Oganov, Glass, Lyakhov and Zhu and featuring local optimization, real-space representation and flexible physically motivated variation operators”.

查阅OUTPUT.txt文件获得最重要的引用。

2.3 错误报告

像其他任何一个如此大型的代码，USPEX在运行过程中可能存在错误。如果在你的模拟中发现异常的运行，你可以将错误的详细，以及INPUT.txt、OUTPUT.txt、log和其他相关文件发送给USPEX的管理员（朱强（alecfans@gmail.com））。

2.4 运行USPEX的机型配置

USPEX可以运行于任何平台—所需要的只是一个能在Linux或者Unix系统环境下运行MATLAB或者OCTAVE软件的CPU。通过使用该系统特殊的远程作业机制，USPEX可以连接在任何远端机上进行计算。

2.5 与USPEX兼容使用的代码

Trial structures generated by USPEX are relaxed and then evaluated by an external code interfaced with USPEX. Based on the obtained ranking of relaxed structures, USPEX generates new structures—which are again relaxed and ranked. Our philosophy is to use existing well-established *ab initio* (or classical forcefield) codes for structure relaxation and energy calculations. Currently, USPEX is interfaced with:

- VASP — <https://www.vasp.at/>
- SIESTA — <http://departments.icmab.es/leem/siesta/>
- GULP — <http://nanochemistry.curtin.edu.au/gulp/>
- LAMMPS — <http://lammps.sandia.gov/>
- DMACRYS — <http://www.chem.ucl.ac.uk/basictechorg/dmacryst/index.html>
- CP2K — <http://www.cp2k.org/>
- Quantum Espresso — <http://www.quantum-espresso.org/>
- FHI-aims — <https://aimsclub.fhi-berlin.mpg.de/>
- ATK — <http://quantumwise.com/>
- CASTEP — <http://www.castep.org/>
- Tinker — <http://dasher.wustl.edu/tinker/>
- MOPAC — <http://openmopac.net/>

The choice of these codes was based on 1) their efficiency for structure relaxation; 2) robustness; and 3) popularity. Of course, there are other codes that can satisfy these criteria, and in the future we can interface USPEX to them.

2.6 如何安装USPEX

当你下载了USPEX的文件之后，需要解压并运行以下命令安装USPEX到用户或者系统级的位置：

```
bash ./install.sh
```

安装者不要求具有root权限。你将会被询问是选择MATLAB还是Octave（如果在系统中发现有Octave程序，安装目录以及确认创建或者使用这个目录。然后系统会提供给你关于环境变量的信息，这项必须设置以确保USPEX在系统中可用，例如：

```
For Bash shell system, add these lines in ~/.bashrc or ~/.profile or /etc/profile:
export PATH=/home/user/bin/USPEX:$PATH
export USPEXPATH=/home/user/bin/USPEX/src
```

```
For C shell system, add these lines in ~/.cshrc or ~/.profile or /etc/profile:
setenv PATH "/home/user/bin/USPEX:$PATH"
setenv USPEXPATH "/home/user/bin/USPEX/src"
```

如果你想改变MATLAB或者Octave的路径，你可以编辑USPEX安装目录里的CODEPATH文件。

2.7 如何运行USPEX

为了运行USPEX，你必须有MATLAB或者Octave，USPEX运行在MATLAB（推荐）或者Octave下，并且在电脑节点上有可运行的外部代码用于结构的弛豫和计算它们的能量(USPEX支持的代码详见章节 ??)。开始你的计算时，根据你想要做什么体系来从示例中找到类似的例子（见附录 9.1），然后从编辑INPUT.txt开始，对变量的编辑详见 4 章节。然后，将外部代码做结构弛豫需要的文件放在Specific/文件夹下，以vasp为例，这样的文件有INCAR_1, INCAR_2, ..., INCAR_N,和POTCAR_A, POTCAR_B, ..., 这里的A, B, ... 指的是与POTCAR文件相对应的化学元素名称。

有两种运行代码的方式—旧的和新的，两种都可以用。

(i)在老的方式中，你的运行文件夹下需要有完整的USPEX代码（USPEX.m文件和FunctionFolder目录等）。然后输入

```
nohup matlab < USPEX.m > log &
```

或者你用的是Octave，输入

```
nohup octave < USPEX.m > log &
```

(ii)新的方式中，如果你使用了基于Python语言的运行器的USPEX安装器，你所需要做的是输入：

```
nohup USPEX -r > log &
```

或者，如果是Octave，输入

```
nohup USPEX -r -o > log &
```

Log文件中包含有模拟过程中信息的记录，如果有错误的话（如果你愿意报告软件的漏洞，请把log文件发送给我们）。

OUTPUT.txt文件包含有每一代计算和分析的详细信息。

对于USPEX运行者，我们有许多人性化的选项：

-v, --version:显示程序的版本号并退出

-h, --help: 显示帮助信息并退出

-p, --parameter: 指明需要获得帮助的参数。如果没有值或者是'all'，则整个INPUT.txt参数都会显示

- e, --example:** 显示USPEX例子的细节。如果没有值或者是'all'，则所有的例子都将被显示。
- c NUM, --copy=NUM:** 复制INPUT文件并指明例子所在文件夹。
- g, --generate:** 产生准备USPEX计算的文件夹，包括AntiSeeds、Seeds、Specific、Submission folders
- r, --run:** 运行USPEX计算
- o, --octave:** 用Octave而不是MATLAB来运行USPEX计算
- clean:** 清理计算文件夹

当进行大规模并行计算时，用户需要做一点工作根据用户的电脑配置文件（因此，我们不敢保证支持解决大规模并行计算的问题）。

有两种提交作业的方式—（1）本地提交；（2）远程提交，取决于你是否将*ab initio*计算任务提交给你运行USPEX和MATLAB相同的机器，或者说你是否将你的工作发给远程的超级计算机。参见关键字*whichCluster*以及 8.9章节。

Please note, that you should have `bash` shell set by default to make USPEX working correctly. Users frequently report issues when running USPEX on a machine with `cs` shell, where “`echo -e ...`” command might not be supported.

2.8 运行USPEX示例：迷你手册

一旦下载了USPEX的软件包并安装了它，你就可以运行第一个USPEX例子。这个例子的详细信息列在附录 9.1：例子列表中。所需要的运行例子（处理例子13）的外部代码显示如下：

- GULP: EX02, EX03, EX08, EX12, EX15 (VC-NEB), EX16
- VASP: EX01, EX07, EX09, EX14 (META)
- LAMMPS: EX04
- ATK: EX05
- CASTEP: EX06
- DMACRYS: EX10
- Tinker: EX11

现在，开始我们的第一个USPEX体验：

2.8.1 测试USPEX Python运行器

为了得到版本信息，我们可以使用以下命令：`>> USPEX -v`

如果USPEX运行正常，你应该得到以下信息。

```
USPEX Version 9.4.2 (19/03/2015)
```

如果不工作，请检查你的安装步骤和系统环境设置，如上所述。

2.8.2 运行例子EX13-3D_special_quasirandom_structure_TiCoO

例子13 (EX13) 不需要任何外部代码，我们可以通过运行这个例子来熟悉USPEX的运行步骤。这个例子的运行大约会花费~30分钟左右的时间。为了开始计算，我们首先创建一个测试文件夹，将示例文件复制过来，然后通过USPEX Python运行器计算，命令如下：

```
>> mkdir EX13
>> cd EX13
>> USPEX -c 13
>> USPEX -r
```

在例子运行的时候，我们现在有时间了解EX13更多的细节。在EX13中，我们通过进化算法优化结构阶数。所以，在INPUT.txt中，我们设置：

```
USPEX : calculationMethod
-4    : optType
```

我们使用以下参数：

```
300   : calculationType
```

```
% atomType
Co Ti O
% EndAtomType
```

```
% numSpecies
16 16 64
% EndNumSpecies
```

为举例说明USPEX研究 $\text{Co}_{16}\text{Ti}_{16}\text{O}_{64}$ 体系。为了减少EX13的计算时间，我们只是作为测试，我们可以减少populationSize和numGenerations这两个参数的值，如：

```
5     : populationSize
5     : numGenerations
```

然而，需要注意的是设置如此小的`populationSize`和`numGenerations`，我们不能期望USPEX找到低能量的结构。由于我们不需要外部代码，我们可以简单设置：

```
% abinitioCode
0
% ENDabinit
```

种子文件放置在`Seeds/POSCARS`—这是初始的 $\text{Ti}_{16}\text{Co}_{16}\text{O}_{64}$ 结构。当你发现文件`USPEX_IS_DONE`时，恭喜你，你已经成功完成了我们的第一个示例。接下来，我们将运行USPEX与外部代码交互的计算。

2.8.3 运行一个与外部代码交互的示例

这一步中，我们建议运行与GULP或者VASP交互的示例，从例子02（EX02）或者例子01（EX01）开始。我们使用USPEX运行器获得EX02的例子信息，创建一个单独的文件夹并复制文件，使用以下命令：

```
>> mkdir EX02
>> cd EX02
>> USPEX -c 2
```

因为在EX02中，我们使用的是GULP，设置

```
% abinitioCode
3 3 3 3
% ENDabinit
```

为了运行一系列的计算而不是单独一个计算，我们应该改变`INPUT.txt`中的以下参数：

```
0      : whichCluster
1      : numParallelCalcs
```

在示例`INPUT.txt`文件中，`whichCluster=QSH`是一个内部的机群名供我们自己的团队使用。我们也提供相同的方式帮助用户定义自己的机群。详细信息参见 8.9。

在`INPUT.txt`中，我们没有指定怎么运行GULP，因为我们假定所有用户使用相同的命令：

```
% commandExecutable
gulp < input > output
% EndExecutable
```

但是要确保这个命令是在你的机子上工作的。如果你想运行EX01，使用的是VASP，你应该这样设置：

```
% abinitioCode
1 1 1 1
% ENDabinit

% commandExecutable
mpirun -np 8 vasp
% EndExecutable
```

如果`commandExecutable`设置错误，启动USPEX计算就会失败。当每个参数都设置正确了，我们就可以通过USPEX运行器运行计算了。使用命令：

```
>> USPEX -r
```

2.8.4 检查结果

开始计算后，你可以检查`results1/`文件夹中的OUTPUT文件。现在，你有了使用USPEX运行简单计算的体验。要想了解更详细的关于USPEX和运行复杂计算的信息，我们可以从接下来章节的介绍中了解。当分析结果时，对结构的可视化是很重要的（可视化方法，详见8.1）。

3 输入与输出文件概述

输入/输出文件依赖于进行结构弛豫的外部代码和作业提交的类型。我们的原理中一个重要的技术要素是多步结构弛豫策略，该方法有坚实的理论基础。为了提供依据能量的正确的结构排列，最后一步的结构和能量必须是高质量的。大多数新产生的结构都距局部最小值很远（例如：所包含的键不是太短就是太长）而它们的高质量弛豫的代价极其昂贵。如果第一步的弛豫使用较为粗略的计算条件，只有最后一步的弛豫才需要高质量的计算，则这种高花费就可以避免。第一步的结构弛豫可以使用代价较低的方式（例如：原子间的势使用GULP）。你可以改变计算条件（基本设置，k点采样网格，赝势或者PAW势）或者近似的程度（原子间势、LDA 或GGA）或者结构弛豫的代码（GULP、DMACRYS、SIESTA、VASP、CP2K、QE、CASTEP 或ATK）对于每个候选结构在结构弛豫期间。更进一步，为了保证结构弛豫的稳健性，也需要多步策略——如果初始力和应力太大，全弛豫可能会导致结构的“爆炸”。因此，我们强烈建议初始优化原子的位置时保证晶格参数为常数，或者优化晶胞的形状和原子位置时保证单位晶胞体积是常数。当体积固定进行优化时，你就不必担心在平面波计算中的Pulay应力。因此，使用小一点的基本设置也可以；然而，对于常压变晶胞弛豫，你需要一个高质量的基本设置。对于结构弛豫，你可以从一个小一点的k点设置开始，但是不要忘记在最后一步结构弛豫中适当增加，以获得精确的能量。

3.1 输入文件

假设计算所在文件目录是~/StructurePrediction，当按照序列运行USPEX，这个目录下将会包含：

- 文件INPUT.txt, 在 4 章节中进行了详细讨论。
- Subdirectory ~/StructurePrediction/Specific/ 和VASP, SIESTA 或者GULP (etc.) 等软件的可执行文件，列出结构弛豫的输入文件— INCAR_1, INCAR_2, ...,和赝势文件。
- Subdirectory ~/StructurePrediction/Seeds 中包含种子文件，其中列出了成分/反成分。注意种子文件应该是VASP5 POSCARS 格式来写入并以POSCARS或者POSCARS_gen (gen (gen是代数) 来对文件进行命名。成分compositions和反成分Anti-compositions文件用来控制在变成分或者单程序段计算中成分的变化。
- Subdirectory ~/StructurePrediction/AntiSeeds—你可以在这里放一些在计算中希望避免产生的特殊的结构，或者不需要明确指定任何结构而使用反种子技术（避免一些在运行中发现的结构）。

3.2 输出文件

输出文件一般在result文件夹里~/StructurePrediction/results1 是用于处处USPEX计

算的结果。如果是一次新的计算，`results2`, `results3`, ... (如果计算被重新启动或运行了几次) 就会把每次计算的结果放在分开的`results*` 文件夹里。当查看`Individuals`里的空间群时，需要注意的是USPEX由于有限的结构弛豫和相对紧的空间群容差，因此经常低估结构的空间群。使用者应该可视化所预测的结构。为了获得确信完整的空间群，可以增大空间群容差（但是这种方式可能很危险），或者以更大的精度对结构进行弛豫。

The subdirectory `~/StructurePrediction/results1`包含以下文件：

- `OUTPUT.txt` — 总结输入变量，由USPEX产生的文件和它们的特征值。
- `Parameters.txt` — `INPUT.txt`文件的复制，供参考
- `Individuals` — 给出每个产生结构的详细信息（能量，单位晶胞体积，空间群，用于产生该结构所做的操作，用于计算能量的 k 点网格，有序度等）。文件`BESTIndividuals`给出每一代最好结构的信息。`Individuals`举例：

Gen	ID	Origin	Composition	Enthalpy (eV)	Volume (Å ³)	Density (g/cm ³)	Fitness	KPOINTS	SYMM	Q_entr	A_order	S_order
1	1	Random	[4 8 16]	-655.062	201.062	4.700	-655.062	[1 1 1]	1	0.140	1.209	2.632
1	2	Random	[4 8 16]	-650.378	206.675	4.572	-650.378	[1 1 1]	1	0.195	1.050	2.142
1	3	Random	[4 8 16]	-646.184	203.354	4.647	-646.184	[1 1 1]	1	0.229	0.922	1.746
1	4	Random	[4 8 16]	-649.459	198.097	4.770	-649.459	[1 1 1]	9	0.128	0.958	2.171
1	5	Random	[4 8 16]	-648.352	202.711	4.662	-648.352	[1 1 1]	2	0.154	1.014	2.148
1	6	Random	[4 8 16]	-643.161	206.442	4.577	-643.161	[1 1 1]	1	0.234	0.946	1.766
1	7	Random	[4 8 16]	-647.678	207.119	4.562	-647.678	[1 1 1]	1	0.224	1.108	2.106
1	8	Random	[4 8 16]	-644.482	203.844	4.636	-644.482	[1 1 1]	1	0.215	0.952	1.857
1	9	Random	[4 8 16]	-647.287	204.762	4.615	-647.287	[1 1 1]	40	0.136	1.142	2.563
1	10	Random	[4 8 16]	-649.459	198.097	4.770	-649.459	[1 1 1]	9	0.128	0.958	2.171

- `convex_hull` — 只用于变成分计算，给出所有热力学稳定的成分，和它们的焓值（每原子）。例如：

```

---- generation 1 ----
10 0 -8.5889
0 14 -8.5893
11 3 -8.7679
---- generation 2 ----
10 0 -8.5889
0 14 -8.5893
11 3 -8.8204
---- generation 3 ----
10 0 -8.5889
0 14 -8.5893
12 4 -8.9945
.....
    
```

- `gatheredPOSCARS` — 弛豫的结构（以VASP POSCAR的格式）。例如：

```

EA1 9.346 8.002 2.688 90.000 90.000 90.000 Sym.group: 1
1.0000
9.346156 0.000000 0.000000
0.000000 8.002181 0.000000
0.000000 0.000000 2.688367
Mg Al 0
4 8 16
Direct
0.487956 0.503856 0.516443
0.777565 0.007329 0.016443
0.987956 0.507329 0.016443
0.277565 0.003856 0.516443
0.016944 0.178753 0.016443
0.019294 0.833730 0.516443
0.746227 0.333730 0.516443
    
```

```

0.748577 0.678753 0.016443
0.516944 0.832431 0.516443
0.519294 0.177455 0.016443
0.246227 0.677455 0.016443
0.248577 0.332431 0.516443
0.416676 0.241774 0.516443
0.559871 0.674713 0.016443
0.205650 0.174713 0.016443
0.348845 0.741774 0.516443
0.613957 0.380343 0.016443
0.804054 0.542164 0.516443
0.113957 0.630842 0.516443
0.304054 0.469021 0.016443
0.848845 0.269411 0.016443
0.705650 0.836472 0.516443
0.059871 0.336472 0.516443
0.916676 0.769411 0.016443
0.651564 0.130842 0.516443
0.461467 0.969021 0.016443
0.151564 0.880343 0.016443
0.961467 0.042164 0.516443
EA2 9.487 4.757 4.580 90.243 90.188 89.349 Sym.group: 1
1.0000
9.486893 0.000000 0.000000
0.054041 4.756769 0.000000
-0.014991 -0.019246 4.579857
Mg Al O
4 8 16
Direct
0.499837 0.633752 0.011361
0.500082 0.131390 0.482012
0.813573 0.257696 0.494520
0.326111 0.625491 0.501746
0.995267 0.254346 0.992293
0.160822 0.689054 0.001270
0.995907 0.760753 0.498354
0.159742 0.192300 0.491958
0.811206 0.761223 0.997857
0.325692 0.125479 0.987935
0.656355 0.695175 0.503322
0.656596 0.199917 0.991605
0.487990 0.763078 0.627771
0.845518 0.645378 0.347890
0.623474 0.895186 0.185946
0.616379 0.395875 0.308861
0.093745 0.991831 0.185467
0.092669 0.494591 0.309957
0.847697 0.118765 0.113434
0.475636 0.251449 0.875207
0.327510 0.787484 0.116764
0.720411 0.975740 0.706398
0.200804 0.880147 0.683027
0.975416 0.612789 0.852917
0.986131 0.108285 0.644081
0.204805 0.364607 0.830780
0.718464 0.496262 0.817031
0.323904 0.257705 0.340590
.....

```

- BESTgatheredPOSCARS—每一代中最好的结构。
- gatheredPOSCARS_unrelaxed—给出弛豫之前产生的结构。
- enthalpies_complete.dat—给出每一步弛豫所得结构的焓值。
- origin—显示某结构是从哪个父代通过何种变化操作得到的。例如：

```

ID Origin Enthalpy Parent-E Parent-ID
1 Random -23.395 -23.395 [ 0]
2 Random -23.228 -23.228 [ 0]
3 Random -23.078 -23.078 [ 0]
4 Random -23.195 -23.195 [ 0]
5 Random -23.155 -23.155 [ 0]
6 Random -22.970 -22.970 [ 0]
7 Random -23.131 -23.131 [ 0]
8 Random -23.017 -23.017 [ 0]
9 Random -23.117 -23.117 [ 0]
10 Random -23.195 -23.195 [ 0]
.....

```

- `gatheredPOSCARS_order` — `gatheredPOSCARS`的内容基本相同，额外给出了每个原子的有序度参量值(Ref.¹⁸)。例如：

```
EA1      9.346  8.002  2.688 90.000 90.000 90.000 Sym.group:  1
1.0000
  9.346156   0.000000   0.000000
  0.000000   8.002181   0.000000
  0.000000   0.000000   2.688367
  Mg   Al   O
  4    8   16
Direct
  0.487956   0.503856   0.516443   1.1399
  0.777565   0.007329   0.016443   1.1399
  0.987956   0.507329   0.016443   1.1399
  0.277565   0.003856   0.516443   1.1399
  0.016944   0.178753   0.016443   1.1915
  0.019294   0.833730   0.516443   1.2474
  0.746227   0.333730   0.516443   1.2474
  0.748577   0.678753   0.016443   1.1915
  0.516944   0.832431   0.516443   1.1915
  0.519294   0.177455   0.016443   1.2474
  0.246227   0.677455   0.016443   1.2474
  0.248577   0.332431   0.516443   1.1915
  0.416676   0.241774   0.516443   1.2914
  0.559871   0.674713   0.016443   1.1408
  0.205650   0.174713   0.016443   1.1408
  0.348845   0.741774   0.516443   1.2914
  0.613957   0.380343   0.016443   1.2355
  0.804054   0.542164   0.516443   1.2161
  0.113957   0.630842   0.516443   1.2355
  0.304054   0.469021   0.016443   1.2161
  0.848845   0.269411   0.016443   1.2914
  0.705650   0.836472   0.516443   1.1408
  0.059871   0.336472   0.516443   1.1408
  0.916676   0.769411   0.016443   1.2914
  0.651564   0.130842   0.516443   1.2355
  0.461467   0.969021   0.016443   1.2161
  0.151564   0.880343   0.016443   1.2355
  0.961467   0.042164   0.516443   1.2161
EA2      9.487  4.757  4.580 90.243 90.188 89.349 Sym.group:  1
1.0000
  9.486893   0.000000   0.000000
  0.054041   4.756769   0.000000
 -0.014991  -0.019246   4.579857
  Mg   Al   O
  4    8   16
Direct
  0.499837   0.633752   0.011361   0.9368
  0.500082   0.131390   0.482012   1.0250
  0.813573   0.257696   0.494520   0.9805
  0.326111   0.625491   0.501746   0.9437
  0.995267   0.254346   0.992293   0.9241
  0.160822   0.689054   0.001270   1.1731
  0.995907   0.760753   0.498354   0.9696
  0.159742   0.192300   0.491958   1.2666
  0.811206   0.761223   0.997857   1.0215
  0.325692   0.125479   0.987935   1.0353
  0.656355   0.695175   0.503322   1.2291
  0.656596   0.199917   0.991605   1.2547
  0.487990   0.763078   0.627771   1.1199
  0.845518   0.645378   0.347890   0.9725
  0.623474   0.895186   0.185946   0.9990
  0.616379   0.395875   0.308861   1.0141
  0.093745   0.991831   0.185467   1.1451
  0.092669   0.494591   0.309957   1.1164
  0.847697   0.118765   0.113434   0.8821
  0.475636   0.251449   0.875207   1.0609
  0.327510   0.787484   0.116764   1.0451
  0.720411   0.975740   0.706398   0.9539
  0.200804   0.880147   0.683027   0.9398
  0.975416   0.612789   0.852917   1.1191
  0.986131   0.108285   0.644081   0.9803
  0.204805   0.364607   0.830780   1.0915
  0.718464   0.496262   0.817031   1.0663
  0.323904   0.257705   0.340590   1.1471
.....
```

- `goodStructures_POSCARS` and `extended_convex_hull_POSCARS` 是一个在v.9版本USPEX可用的文件。这个文件非常便于分析，它以稳定性递减的顺序展示所有不同的结构，从最稳定的结构开始，到最不稳定的。

- `compositionStatistic` 包含有该成分是由哪几种变化操作产生的信息。例如：

Comp/Ratio	Total	Random	Heredity	Mutation	Seeds	COPEX	Best/Convex
[0.0000 1.0000]	30(63)	21	8	1	0	0	33
0 8	4(4)	2	2	0	0	0	0
0 9	2(2)	2	0	0	0	0	0
0 10	5(5)	3	2	0	0	0	0
0 11	2(2)	0	1	1	0	0	0
0 12	6(35)	6	0	0	0	0	29
0 13	2(3)	1	1	0	0	0	1
0 14	0(0)	0	0	0	0	0	0
0 15	2(2)	2	0	0	0	0	0
0 16	5(8)	5	0	0	0	0	3
0 3	1(1)	0	1	0	0	0	0
0 4	1(1)	0	1	0	0	0	0
[0.1250 0.8750]	52(52)	32	9	11	0	0	0
1 7	20(20)	4	8	8	0	0	0
2 14	32(32)	28	1	3	0	0	0
[0.1111 0.8889]	25(25)	9	14	2	0	0	0
1 8	25(25)	9	14	2	0	0	0
[0.1000 0.9000]	16(16)	9	5	2	0	0	0
1 9	16(16)	9	5	2	0	0	0
[0.0909 0.9091]	11(11)	5	4	2	0	0	0
1 10	11(11)	5	4	2	0	0	0
[0.0833 0.9167]	17(17)	8	2	7	0	0	0
1 11	14(14)	8	2	4	0	0	0
2 22	3(3)	0	0	3	0	0	0

- graphical files (*.pdf) — 快速分析结果:

- `Energy_vs_N.pdf` (`Fitness_vs_N.pdf`) — 能量（适应度）与结构序号的函数；
- `Energy_vs_Volume.pdf` — 能量与体积的函数；
- `Variation-Operators.pdf` — 子代与亲代的能量对比；不同的颜色对应不同的运算方法（此图可用于评估不同的变异运算的结果）；
- `E.series.pdf` — 弛豫的第 i 步和第 $i + 1$ 步的能量关联；有助于发现问题和改进弛豫文件输入。
- 对于可变成分有另外的图`extendedConvexHull.pdf`，用于显示形成焓与组分的函数。
- `compositionStatistic.pdf` — visualization of `compositionStatistic` file.

3.3 Specific/ 文件夹

用于结构弛豫的可执行文件和列举的输入文件应该放在`~/StructurePrediction/Specific/`文件夹下，对于不同的交互代码，例如VASP, SIESTA, GULP, *etc.*都应如此。

- 对于VASP，文件`INCAR_1`, `INCAR_2`, ..., *etc.* 等定义了弛豫和能量计算在每一步的弛豫中应该怎样实施（我们推荐至少三步），相应的`POTCAR_1`, `POTCAR_2`文件是赝势文件。例如：`INCAR_1`, `INCAR_2`对于原子位置和晶胞常数在体积固定的情况下进行了非常粗略的结构弛豫，`INCAR_3`在恒定外压下以中等精度进行充分结构弛豫，`INCAR_4`则进行精确的计算。每一个高一级的结构弛豫都是从低一级的优化结果开始并且改进它们。`POTCAR`文件可以仅仅只是将相关元素文件放在`Specific/`文件夹下而被选择性定义。比如`POTCAR_C`, `POTCAR_O`, *etc.*

- 对于SIESTA, 你需要赝势文件和输入文件, , ...
- 对于GULP, 文件goptions_1, goptions_2, ... 和ginput_1, ginput_2... 必须有。前者指定哪种优化将被实施, 后者指定细节 (原子间势、压力、温度和优化循环数等)。
- For DMACRYS, fort.22 is the file for general control parameters. The classical force field is given by the file of fit.pots. File cutoff defines the maximum bond length of the intra-molecular bonds.
- 对于CASTEP, 结构文件由cell_1, cell_2, ... 给出, 而计算参数由param_1, param_2, ... 给出, 相应的赝势文件也必须有。
- 对于CP2K, 文件cp2k_options_1, cp2k_options_2, ..., 是必须准备好的。所有的文件都应该是含有所有除了原子坐标和晶胞参数 (这些会被USPEX写在最后一行“&END FORCE_EVAL”) 的所有参数的正常CP2K输入文件。因为程序会从USPEX-1.cell和USPEX-pos-1.xyz文件中读取输出文件, 所以“name of the project”应该一直是USPEX。我们建议是进行弛豫要至少三步跟VASP类似) ——第一次优化是固定晶格只优化原子位置, 之后的都进行优化。
- 对于Quantum Espresso, qEspresso_options_1, 文件qEspresso_options_2, qEspresso_options ..., 是必须准备的。所有的文件都应该是含有所有除了原子坐标、晶胞参数和k点 (这些会被USPEX写在最后的文件里) 的所有参数的正常QE输入文件。我们建议进行多步弛豫。例如, qEspresso_options_1只是做了固定晶胞参数弛豫和原子位置的粗略计算, qEspresso_options_2是中等精度条件下的加常外压的全结构弛豫, qEspresso_options_3就是高精度的计算。

3.4 Specific/文件夹里的INCAR_*文件 (VASP)

为了正确运行USPEX, 有一些对Specific/文件夹中文件的提示, 以便使用USPEX进行结构弛豫。我们以VASP为例子:

- 计算产生的最终结构必须被很好的弛豫, 能量和精度收敛。要点是你设置的能量排名必须正确 (要检查这个, 查看输出中的E_series.pdf文件)。
- 计算的POTCAR文件: 为了得到正确的结果, 你的赝势的核 (或者PAW势) 不应该重叠超过10–15%。
- 为了以一个低的代价获得精确的弛豫, 对每一个结构设置至少三步弛豫。就是说至少有三个INCAR文件(INCAR_1, INCAR_2, INCAR_3, ...)。通常我们会设4–5步弛豫。
- 你的初始结构通常远离局部最小点, 在这种情况下, INCAR_1,2 首先应该在保持体积固定的情况下 (ISIF=4) 弛豫原子和晶胞形状, 然后在INCAR_3,4中做完全弛豫 (ISIF=3), 在INCAR_5中完成非常精确的单点计算 (ISIF=2, NSW=0)。

例外情况：当你做固定晶胞预测，和在进化分子动力学中（除了全弛豫），你必须设ISIF=2。

- 当你的体积不改变时，你可以使用默认的平面波截止能。当你再INCAR中使用ISIF=3，你必须将其增加30–40%，否则你会得到一个大的Pulay 应力。你的收敛准则一开始比较松，但是在结束时应该紧一点，例如：EDIFF=1e-2和EDIFFG=1e-1在INCAR_1中，逐渐的收紧至EDIFF=1e-4和EDIFFG=1e-3在INCAR_4中。timesteps(N最大值应该足够大保证好的弛豫，但也不能太大以避免在一些差的构型上浪费太多机时。你的系统越大，NSW就越大。
- 选择一个有效的弛豫算法可以节省很多时间，在VASP中，我们推荐开始弛豫时使用共轭梯度（IBRION=2, POTIM=0.02）而当结构接近局部小值时，转换至IBRION=1, POTIM=0.3。
- 即使你研究的是绝缘体系，你采样的许多构型将会是金属性的，所以为了得到收敛的结果，你必须使用“金属化”处理—这对于金属和绝缘体同样适用。我们推荐Methfessel-Paxton重叠方案(ISMEAR=1)。对于一个清楚的金属系统，使用ISMEAR=1, SIGMA=0.1–0.2。对于一个清楚的绝缘系统，我们推荐ISMEAR=1和SIGMA从0.1 (INCAR_1) 开始降至0.03–0.04。

这里，我们提供一个INCAR文件的例子，这是关于单元晶胞中有16个原子的C，默认的POTCAR中ENCUT=400 eV：

```

INCAR_1:                INCAR_2:                INCAR_3:
PREC=LOW                PREC=NORMAL            PREC=NORMAL
EDIFF=1e-2              EDIFF=1e-3            EDIFF=1e-3
EDIFFG=1e-1            EDIFFG=1e-2           EDIFFG=1e-2
NSW=65                  NSW=55                 ENCUT=520.0
ISIF=4                  ISIF=4                 NSW=65
IBRION=2                IBRION=1               ISIF=3
POTIM=0.02              POTIM=0.30            IBRION=2
ISMEAR=1                ISMEAR=1               POTIM=0.02
SIGMA=0.10              SIGMA=0.08             ISMEAR=1
                        SIGMA=0.07

INCAR_4:                ISMEAR=1
PREC=NORMAL              SIGMA=0.06
EDIFF=1e-4
EDIFFG=1e-3
ENCUT=600.0
NSW=55
ISIF=3
IBRION=1
POTIM=0.30

                        INCAR_5:
                        PREC=NORMAL
                        EDIFF=1e-4
                        EDIFFG=1e-3
                        ENCUT=600.0
                        NSW=0

```

```
ISIF=2          ISMEAR=1
IBRION=2        SIGMA=0.05
POTIM=0.02
```

除了在准进化过程中我们没有改变晶胞的形状，准分子动力学的原理其实和USPEX非常类似。因此我们需要给所有的准进化添加ISIF=2。如果开启了全弛豫模式，我们可以再所有的步的弛豫中添加ISIF=3。如果我们按以下方式设置：The philosophy of METADYNAMICS is very similar to USPEX, except that we DO NOT change the cell shape during the META evolution. Therefore, we need to put ISIF=2 for all META steps. If the full relaxation mode is on, we can put ISIF=3 for the steps of full relaxation. Therefore, if we have the following set up:

```
% abinitioCode
1 1 1 (1 1)
% ENDabinit
```

那么ISIF应该是“2 2 2 3 3”对应的是INCAR_1, ..., INCAR_5。

不同于USPEX，VC-NEB方法并不需要来自外部代码的结构弛豫，它通过VC-NEB自身并结合外部代码计算得到的力来进行结构弛豫。因此设置文件时会有一些不同。以VASP的INCAR文件为例，我们需要设置NSW=0去避免结构弛豫，但是ISIF=2或者3来提取原子上的力和晶格上的应力张量。我们也建议设置PREC=Accurate从而对力和应力有一个好的估计以加速VC-NEB的计算。一个针对VC-NEB的INCAR文件例子如下：

```
INCAR_1:
PREC=Accurate
EDIFF=1e-4
EDIFFG=1e-3
ENCUT=600.0
NSW=0
ISIF=2
IBRION=2
POTIM=0.02
ISMEAR=1
SIGMA=0.05
```

4 输入文件INPUT.txt 的设置

典型的INPUT.txt文件在附录 9.3中给出。下面我们介绍输入中最重要的参数。许多参数都有可靠的默认值，当你在输入文件中遗漏掉它们时，这些默认值就变得有用（这就允许用户可以有非常短的输入文件！）。那些没有默认值选择的参数就需要用户根据具体的体系进行设置。可以<http://han.ess.sunysb.edu> 上查阅在线工具，这些小工具可以帮助用户更好的准备INPUT.txt文件，分子文件以及分析一些结果。手册的章节 7简要地讨论了这些工具用法。

4.1 运行的类型和系统

▷ *variable* `calculationMethod`

Meaning: 明确计算方法

可能值（特性）:

- USPEX— 晶体结构预测的发展算法
- META—进化准动态算法
- VCNEB — 用变晶胞微动弹性带方法探索过渡态路径
- PSO — 修正的PSO算法
- TPS — 过渡路径抽样方法（迄今未发布）
- MINHOP — 最小值跳跃方法（迄今未发布）
- COPEX — 另一种新技术，不久将发布

Default: USPEX

Format:

```
USPEX : calculationMethod
```

▷ *variable* `calculationType`

Meaning: 也就是不论是体晶、纳米团簇还是表面结构，都可以被预测。这种可变因素由三部分组成，包括维度、分子状态和成分的可变性，以及“s” or “S”表示的自旋选项:

- 维度:
 - “3” — 体材料
 - “2” — 表面, “-2” — 二维晶体

- “1” — 聚合物
- “0” — 纳米粒子

- 分子状态:

- “0” — 无分子
- “1” — 分子计算

- 化学组成的变化:

- “0” — 定成分
- “1” — 变成分

Default: 300

Format:

301 : calculationType

注意: 如果calculationType=310, 也就是, 预测分子晶体, 然后USPEX希望你提供含有所有的分子类型的分子几何体的MOL_1, MOL_2, ... 这些分子将被安置在新的几何结构中。有效的选择: 300 (s300), 301 (s301), 310, 000(s000), 200(s200), 201(s201), -200(-s200), (还有迄今未发布的: 110, 311)

▷ *variable* **optType**

Meaning: 用户可以通过这个值来定义想要最大化或者最小化的性质。默认值是最小化焓和体积或者最大化optTypes剩下的性质设置-但是你可以清楚地定义你想要最大化或者最小化的性质。你也可以用某个目标值来优化性质 (例如, 对太阳能光伏来说带隙接近于1.34 eV的材料是很有吸引力的)。

可能的值(特征):

Value	Number	Description
enthalpy	1	为了发现稳定相
volume	2	体积最小化 (找到最密结构)
hardness	3	硬度最大化 (找到最硬相)
struc_order	4	有序度最大化 (发现最有序结构)
aver_dist	5	在一代内结构差别的平均值的最大化
diel_sus	6	静态电介质极化率最大化 (仅适用于VASP和GULP)
gap	7	带隙最大化 (仅适用于VASP)
diel_gap	8	电能存储能量最大化 (仅适用于VASP)
mag_moment	9	磁化强度最大化 (仅适用于VASP)
quasientropy	10	结构准熵值最大化

与弹性力学相关的性质 (“11**”):

值	序号	说明
K, 体积弹性模量	1101	体弹性模量最大化
G, 剪切模量	1102	剪切模量最大化
E, 杨氏模量	1103	杨氏模量最大化
ν , 泊松比	1104	泊松比最大化
G/K, Pugh模量比	1105	Pugh模量比最大化
Hv, 维氏硬度	1106	维氏硬度最大化
Kg, 断裂韧性	1107	断裂韧性最大化
D, 德拜温度	1108	德拜温度最大化
Vm, 声速	1109	声速最大化
S-波速	1110	S-波速最大化
P-波速	1111	P-波速最大化

注意: 与弹性力学相关的性质仅仅适用于VASP (从VASP5.1开始)和GULP。对于VASP用户, 需要在Specific/文件夹中增加至少一个INCAR_* 文件, INCAR_*文件应该包括的参数有IBRION=6, ISIF \geq 3和NFREE=4。体积弹性模量剪切模量和杨氏模量的估计值是Voigh-Reuss-Hill (VRH) 的平均值。维氏硬度是用Chen-Niu模型¹⁹计算得到。断裂韧性的优化用理论断裂韧性的最低值作为适用值。

Default: enthalpy

Format:

enthalpy : optType

注意:

(1) 如果想要反向优化, 需添加负号。例如: 要使静态介电系数最小化, 赋上“-diel_sus”。

(2) 如果optType=gap (带隙) 或diel_gap (电介质带隙), 这里的带隙我们使用的是一个扩展函数, 这个扩展函数对于金属表现出连续性。—也就是说, $\Delta E_g - g(E_F)/N$, ΔE_g 是带隙, $g(E_F)$ 是在费米能级处的态密度 (对金属而言), 而N是单胞内的原子数。由于这个扩展函数的连续性, 与带隙相关量的全局最大化甚至可以用于解决金属的问题。对金属而言, 这个值等于费米能级处的态密度。对于半导体和绝缘体而言这个值等于带隙。

Fig. 5是以硬度为目标优化函数搜索TiO₂硬度最大的结构 (选择类型=硬度, 显示了可能的最大硬度为14GPa²⁰, 这是对Dubrovinsky (2001) 关于TiO₂的超硬度的反驳²¹。这是一个很好的例子: 一个简单的USPEX就可以解决长期存在的争论。

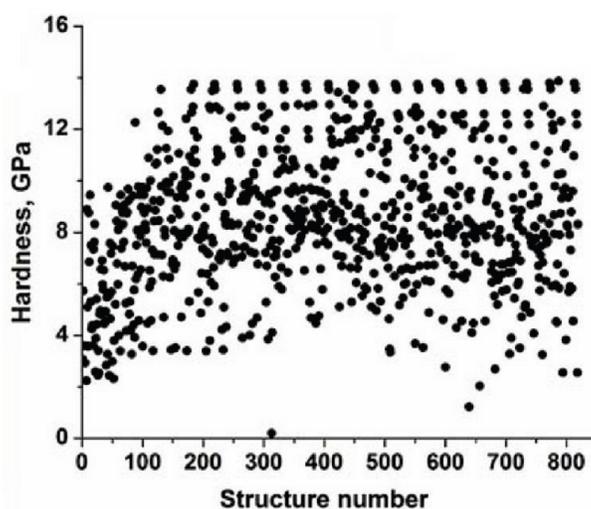


Figure 5: 预测最硬的TiO₂.

现在, 你需要明确所要计算体系的信息。

▷ variable atomType

Meaning: 描述每种原子类型的特性。

Default: 无, 必须明确给出

Format:

如果你更喜欢用门捷列夫元素周期表的原子序数, 设置:

```
% atomType
12 14 8
% EndAtomType
```

Or, if you prefer to use atomic names, specify:

```
% atomType
```

```
Mg Si O
% EndAtomType
```

或者，你更喜欢用元素名，设置：

```
% atomType
Magnesium Silicon Oxygen
% EndAtomType
```

▷ *variable* `numSpecies`

Meaning: 描述每种类型原子的数目。

Default: 无，必须明确给出

Format:

```
% numSpecies
4 4 12
% EndNumSpecies
```

这表示第一种类型有4个原子，第二种类型有4个原子，第三种类型有12个原子。

Notes: 对于变成分的计算，你需要列出如下所示的组成构建块：

```
% numSpecies
2 0 3
0 1 1
% EndNumSpecies
```

这表示第一个构建块的分子式是 A_2C_3 ，第二个构建块的分子式是 BC ， A 、 B 、 C 在原子类型块中已经解释过。所有结构都满足公式 $xA_2C_3 + yBC$ with $x, y = (0,1,2,\dots)$ — 或者是 $A_{2x}B_yC_{3x+y}$ 。如果你想做 A - B - C 体系中所有可能组成的预测，应该这样设置：

```
% numSpecies
1 0 0
0 1 0
0 0 1
% EndNumSpecies
```

你也可以用一个化学式的计量比数来做定成分计算，这种情况下，只需要设置一行 (`calculationType=301`) 如下为 A_2BC_4 化合物的例子：

```
% numSpecies
2 1 4
% EndNumSpecies
```

▷ *variable* `ExternalPressure`

Meaning: 指定你所要计算体系的外部压力，单位是GPa。

Default: 0

Format:

100 : ExternalPressure

Note: 从USPEX 9.4.1 起, 压力的值(单位GPa)可以直接在INPUT.txt文件中设置。不需要再弛豫优化文件Specific/再次指定。

▷ variable **valences**

Meaning: 描述每类原子的化合价。这仅用于评估结合的硬度, 而结合硬度又用来计算近似的动力学矩阵(对软模变异)和晶体的硬度。

Default: USPEX有一系列的系统默认化合价(见附录 9.8)。然而, 需要注意的是, 对一些元素(例如N、S、W、Fe和Cr等), 有许多可能的化合价。除非你要计算硬度, 否则这没有问题, 你可以用系统默认化合价。如果你要做硬度计算, 你必须明确给出化合价。

Format:

```
% valences
2 4 2
% EndValences
```

▷ variable **goodBonds**

Meaning: 在矩阵中指定两原子间最低的键价被认为是最重要的键价。就像离子距离矩阵IonDistances (如下), 这是一个上三角形式的方阵。这仅用来计算硬度和软模变异。我们可以用公式 $goodBonds = \frac{valence}{max_coordination_number}$ 。

Default: USPEX可以给出goodBonds合理的默认估值, 你可以在输出文件OUTPUT.txt中得到这些数值。对大多数情况, 用默认值已经足够, 但是对于硬度的计算, 你必须仔细地审查这些数值, 或者手动设置。更多细节, 请参考附录 9.9

Format:

```
% goodBonds
10.0 10.0 0.2
0.0 10.0 0.5
0.0 0.0 10.0
% EndGoodBonds
```

注意: 矩阵的维度必须与原子种类数或原子名称数相等。如果只有一种原子, 那么矩阵就填充该数字。上面的矩阵解读如下: 可被认为是一个键, Mg-Mg间的距离要足够短使其键价为10或者更大, Mg-Mg键的距离必须足够短。对Mg-Si键, Si-Si键和O-O键也有同样要求(通过使用这种专一的标准, 我们有效地排除了来自软模变异和硬度计算的交互作用), 然而在用于硬度和软模变异计算的Mg-O键的键价取为0.2或者更大, Si-O键的键价取为0.5或者更大。

▷ variable **checkMolecules**

Meaning: 打开或关闭原始分子的再弛豫 (MOL_1, MOL_2, ...) 是否完整。可用于分子晶

体(`calculationType=310, 311`)。

可能的数值（整数）：

- 0 — 未经核实，被认为是损坏了的结构或是合并的分子（我们强烈建议不使用这种类型）。
- 1 — 执行检查命令，所有损坏或是合并的分子均被遗弃。

Default: 1

Format:

1 : `checkMolecules`

▷ *variable* `checkConnectivity`

Meaning: 打开或关闭硬度计算和在`softmutation1`里连接相关的标准。

可能的值(整数)：

- 0 — 未经检查, 没有硬度计算；
- 1 — 没有考虑连接相关, 有硬度计算。

Default: 0

Format:

1 : `checkConnectivity`

4.2 种群

▷ *variable* `populationSize`

Meaning: 每一代中的结构数。如果需要的化，第一代可以单独设置。

Default: 当 N 是晶胞中的原子数时（或者是变成分结构中的`maxAt`最大原子数）， $2 \times N$ 四舍五入最接近10，那么上限值是60。通常可以依靠系统默认设置。

Format:

20 : `populationSize`

▷ *variable* `initialPopSize`

Meaning: 第一代中的结构数。

Default: 等于`populationSize`.

Format:

20 : initialPopSize

Note:在许多情况下，我们建议这两个参数值相等。有时（尤其是在变成分计算中）指定initialPopSize值大于initialPopSize值更为有用。设置一个很小的初始群也是可能的，当你想要完全从几个种子结构产生第一代时这样设置就变得很有用。

▷ variable numGenerations

Meaning: 最大模拟代数。当stopCrit的最优结构保持不变时，模拟可以较早地结束。

Default: 100

Format:

50 : numGenerations

▷ variable stopCrit

Meaning: 当停止标准代数中的最优结构不再变化，或者是代数已满的话—无论哪个先发生，模拟就会停止。

Default: 定成分运行的总原子数，变成分运行的最大原子数maxAt。

Format:

20 : stopCrit

4.3 适者生存和选择

▷ variable bestFrac

Meaning: 目前这一代中用于产生下一代的比例。

Default: 0.7

Format:

0.7 : bestFrac

注意: 这是一个非常重要的参数，值在0.5–0.8之间比较合理。

▷ variable keepBestHM

Meaning: 定义有多少个最优结构将会在下一代中存活。

Default: 0.15×populationSize

Format:

3 : keepBestHM

▷ variable reoptOld

Meaning: 定义存活的结构是否需要优化. 如果`reoptOld=0`, 这些结构不会在下一代中重新优化; 如果`reoptOld=1`, 这些结构将会在下一代中重新优化。通常设定`reoptOld=0`比较合理

Default: 0

Format:

1 : reoptOld

4.4 结构产生和变异操作

▷ *variable symmetries*

Meaning: 晶体可能的空间群, 二维晶体/表面的平面群组, 或者是团簇的点群。在该列表中用随机选择的对称群, 用随机生成的晶格常数和原子坐标产生确定结构数。在这个过程中, 从普通位置上将生成特殊的威科夫 (Wyckoff) 位置(Fig. 6)。

Default:

- 对三维晶体: 2-230
- 对二维晶体/表面: 2-17
- 团簇: E C2 D2 C4 C3 C6 T S2 Ch1 Cv2 S4 S6 Ch3 Th Ch2 Dh2 Ch4 D3 Ch6 O D4 Cv3 D6 Td Cv4 Dd3 Cv6 Oh Dd2 Dh3 Dh4 Dh6 Oh C5 S5 S10 Cv5 Ch5 D5 Dd5 Dh5 I Ih

Format:

```
% symmetries
195-198 200 215-230
% EndSymmetries
```

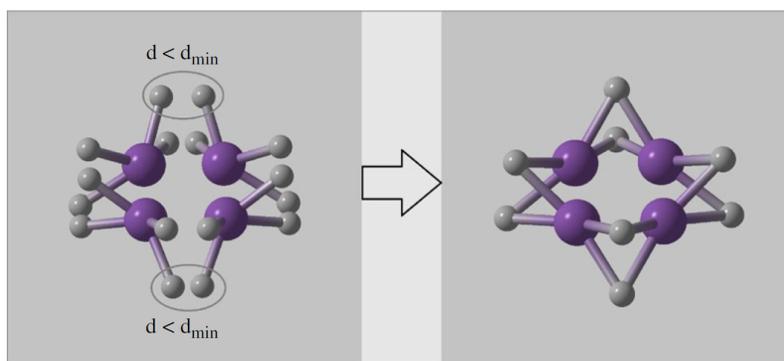


Figure 6: 整合原子到特殊威科夫位置的例子(来自Ref.¹³).

▷ variable `fracGene`

Meaning: 通过遗传所产生的结构所占的百分比; 0.1 means 10%, *etc.*

Default: 0.5

Format:

0.5 : `fracGene`

▷ variable `fracRand`

Meaning: 通过空间群随机产生的结构所占的百分比。

Default: 0.2

Format:

0.20 : `fracRand`

▷ variable `fracPerm`

Meaning: 通过置换所产生的结构所占的百分比; 0.1 means 10%, *etc.*

Default: 0.1 如果有超过一种类型的原子/分子; 0 其他情况。

Format:

0.1: `fracPerm`

▷ variable `fracAtomsMut`

Meaning: 指定通过软模变异或交换所产生的机构所占的百分比。

Default: 0.1

Format:

0.1 : `fracAtomsMut`

注意: 你可以通过指定`softMutTill`用软模变异或协同变异。

▷ variable `fracRotMut`

Meaning: 通过分子取向变异所产生的结构所占的百分比。0.1代表10%，等等; 0.1 means 10%, *etc.*

Default: 0.1 对于分子晶体; 0 其他情况

Format:

0.1 : `fracRotMut`

▷ variable `fracLatMut`

Meaning: 由晶格变异所获得的结构的百分比。0.1代表10%，等等。; 0.1 means 10%,

etc.

Default: 0 固定晶胞预测; 0.1 其他情况

Format:

```
0.1 : fracLatMut
```

注意: 如果所有分数的总和(`fracGene + fracRand + fracPerm + ...`) 不等于1,它们将会被重新调节。

▷ *variable* `howManySwaps`

Meaning: 对置换来说, 互换原子对的数量由在1和`howManySwaps` 之间的统一分配所随机决定。

Default: $0.5 \times (\text{可能的互换原子对的最大值})$. 如果原子 Na 和 Nb , 原子 Nc 和 Nd 发生了互换, 那么可能的互换原子对的总数是 $\min(Na, Nb) + \min(Nc, Nd)$, 并且`howManySwaps`的默认设置为 $0.5 \times [\min(Na, Nb) + \min(Nc, Nd)]$ 。在许多情况下, 采用默认值是一个不错的选择。

Format:

```
5 : howManySwaps
```

▷ *variable* `specificSwaps`

Meaning: 明确在置换中允许哪种原子类型发生互换。

Default: 空行, 这意味着没有明确的互换, 所有的原子都发生了置换。

Format:

```
% specificSwaps
1 2
% EndSpecific
```

注意: 在这种情况下, 第一种类型的原子可以与第二种类型的原子发生互换。如果你想要尝试所有可能的互换, 在该关键块里留一空行, 或者删除这一部分。

▷ *variable* `mutationDegree`

Meaning: 在软模变异的位移的最大值的单位是Å. 设置了软模变异和协同变异的最大位移矢量使其最大位移等于`mutationDegree`。

Default: $3 \times (\text{平均原子半径})$

Format:

```
2.5 : mutationDegree
```

▷ *variable* `mutationRate`

Meaning: 晶格变异的应变矩阵的标准偏差的极小值。应变矩阵部分从高斯分布中随机选择，并且只允许取-1到1之间的值。晶格变异的本质上是把准动力学的思想融入到了我们的方法中^{6;22}，新的结构是由一些异质结构的晶胞扭曲产生的。不像在准动力学中，在我们的方法中，变形是不累积的，所以弹性部分可以足够大去获得新结构。

Default: 0.5

Format:

0.5 : mutationRate

晶格变异与较弱的软模变异结合在一起时一个好选择:

▷ *variable* **DisplaceInLatmutation**

Meaning: 将软模变异指定为晶格变异的一部分，并且设置最大位移的量级为Å.

Default: 1.0

Format:

1.0 : DisplaceInLatmutation

▷ *variable* **AutoFrac**

Meaning: 程序根据计算的演变，自动分配对变量算符所占的百分比（参数控制）。这使得计算速度加快了两倍左右。当要选择用户自定义变异操作时，设置AutoFrac=0。

Default: 0

Format:

1 : AutoFrac

4.5 约束条件

相同的结构可以再坐标系中以无限的方式展现出来（“模数不变性”）。这些大量相同的选择将会导致非常扁平的晶胞单元，从而为结构弛豫和能量计算带来问题（例如需要很大数量的 k 点）。约束条件在结晶学中已经熟知，那就是晶胞角度在 60° 到 120° 之间，不会移出任何一个多余的和有问题的晶胞（例如：晶胞中允许的 $\alpha = \beta = \gamma \sim 120^\circ$ 实际上是扁平的）。因此我们发明了一种特殊的方法：用最短的晶格矢量来获得特殊的晶胞形状^{23;24}。如果至少有一个晶格矢量的投影在其他的晶格矢量上或者在对立面上的斜的晶格矢量的长度大于（通过模比较）该晶格矢量长度的一半，就会发生变形。也就是说，对于 \mathbf{a} 和 \mathbf{b} 电子对，或者是 \mathbf{c} 和 $(\mathbf{a} + \mathbf{b})$ 电子对，有以下标准：

$$\frac{|\mathbf{a} \cdot \mathbf{b}|}{|\mathbf{b}|} > \frac{|\mathbf{b}|}{2} \quad (2)$$

$$\frac{|\mathbf{a} \cdot \mathbf{b}|}{|\mathbf{a}|} > \frac{|\mathbf{a}|}{2} \quad (3)$$

$$\frac{|\mathbf{c} \cdot (\mathbf{a} + \mathbf{b})|}{|\mathbf{c}|} > \frac{|\mathbf{c}|}{2} \quad (4)$$

$$\frac{|\mathbf{c} \cdot (\mathbf{a} + \mathbf{b})|}{|\mathbf{a} + \mathbf{b}|} > \frac{|\mathbf{a} + \mathbf{b}|}{2} \quad (5)$$

例如，在标准2中，新的矢量 \mathbf{a}^* 值等于：

$$\mathbf{a}^* = \mathbf{a} - \text{ceil}\left(\frac{|\mathbf{a} \cdot \mathbf{b}|}{|\mathbf{b}|^2}\right) \text{sign}(\mathbf{a} \cdot \mathbf{b})\mathbf{b} \quad (6)$$

这种变形迭代进行，完全避免了劣质的晶胞形状，从而解决了问题。在该变形中，原子坐标部分发生了改变，以保证初始结构和改变后的结构是完全相同的（在变形过程中，原子的笛卡尔坐标保持不变）。

▷ *variable* `minVectorLength`

Meaning: 设置最新产生的结构的晶胞参数的最小长度。

Default: $1.8 \times$ 最大原子的共价直径。对于分子晶体 (`calculationType = 310, 311`) 默认最大值是 $1.8 \times \text{max}(\text{MolCenters})$ 。

Format:

2.0 : minVectorLength

当原子间距离足够小时，通常使用的计算方法（如赝势，PAW，LAPW和其他参数化的力场）将不会起作用。这种情况需要避免，你可以用上三角形形式的离子距离方阵 `IonDistances` 指定每对原子之间的最小距离：

▷ *variable* `IonDistances`

Meaning: 在不同的原子类型之间设置最小原子距离矩阵。小于离子距离的数值在物理意义上是没有意义的，需要严格避免。

Default: A和B原子之间的离子距离被估计为 $0.22 \times (V_A^{1/3} + V_B^{1/3})$ ，但是不会超过 1.2 \AA ，在分子类晶体计算中是 $0.45 \times (V_A^{1/3} + V_B^{1/3})$ ，此处 V_A 和 V_B 是USPEX中所估计得到的A原子和B原子的默认体积值。

Format:

```
% IonDistances
1.0 1.0 0.8
```

```

0.0 1.0 0.8
0.0 0.0 1.0
% EndDistances

```

注意： 矩阵的大小必须等于原子种类数。如果上面例子中的化合物是 MgSiO_3 ，那么矩阵应该按如下解释：在一个新产生的结构中允许的Mg-Mg 距离为 1.0 \AA ，Mg-Si距离、Si-Si距离和O-O距离也是 1.0 \AA ，而最小的Mg-O距离和Si-O距离是 0.8 \AA 。你可以把特定体系的更多信息放进这个键长矩阵，举例来说，如果你知道Mg原子倾向离得很远，在你的体系中最近会小于 3 \AA ，你可以列入该信息。然而需要注意的是最小距离越大，计算产生符合这些约束条件的结构就越困难(特别是大体系)。所以尽量折中，并且谨记离子距离 必须远远小于实际的键长。

▷ variable `constraint_enhancement`

Meaning: 可以用IonDistances矩阵（通过变为constraint_enhancement的倍数）来严格约束计算产生的对称随机结构（适用于所有变异操作，未增强的离子距离IonDistances矩阵依然可以应用）。允许你应用离子距离矩阵中严格的约束条件（通过乘以增强约束条件）。确保你在知道自己正在用的是什么意义的时候使用。

Default: 1

Format:

```
1 : constraint_enhancement
```

对分子晶体，下面的关键块体非常重要：

▷ variable `MolCenters`

Meaning: 分子中心之间的最小距离矩阵。任何小于那些明确给出的分子的大重叠的距离，在物理意义上都是不合理的，需要严格避免。

Default: 零矩阵适用于非分子计算，分子晶体没有默认（必须明确给出）。

Format:

```

% MolCenters
5.5 7.7
0.0 9.7
% EndMol

```

注意： 在上面的例子中，有两种分子类型。在所有产生的结构中，第一种类型的分子的几何中心之间的距离必须至少为 5.5 \AA （A-A距离），第一种类型和第二种类型分子几何中心之间的距离— 7.7 \AA （A-B距离），第二种分子几何中心之间的距离— 9.7 \AA （B-B距离）。

4.6 晶胞

在结构弛豫之前，指定在给定条件下的单胞体积对产生所有新结构是很有用的。这个指定的初始体积可以再Latticevalues这一项中输入：

▷ *variable* `Latticevalues`

Meaning: 明确给出单胞或抑制晶格参数的初始体积。

Default: 对于晶胞体积不需要明确给出值—USPEX有一个强大的计算体系可以给出任何压力下的合理的估计值。

Format:

```
% Latticevalues
125.00
% Endvalues
```

Notes: (1) 该体积仅用于初始的猜测，并且只影响第一代，每个结构都经过充分优化，且采用与最低（自由）能相对应的体积。这个关键部分还有其他用途：当你知道晶格参数时（例如从实验中），你就可以在晶格参数`Latticevalues`中明确指定，而不是使用单胞体积，例如：

```
% Latticevalues
7.49 0.0 0.0
0.0 9.71 0.0
0.0 0.0 7.07
% Endvalues
```

你也可以通过列出 a 、 b 、 c 、 α 、 β 和 γ 值来指定晶胞参数：

```
% Latticevalues
10.1 8.4 12.5 90.0 101.3 90.0
% Endvalues
```

注意：如果你要做定成分的单斜晶胞的计算，则要做专门的 β 角度设置（标准设置）。

(2) 对于变成分计算，需要明确给出计算搜索空间最后一个的体积。例如：

```
% Latticevalues
12.5 14.0 11.0
% Endvalues
```

(3) 用户不需要给出晶格常数关键子块中单胞或原子的体积`Latticevalues`—一个专门的算法已经被编写进USPEX中来对给定压力下体系的体积进行精确估计，因此不需要用户专门去设定它。这个选项很好地适用于需要在INUPT中设置体积的所有计算类型`calculationType: 3**, 2D-crystals, 110, 000`，你也可以用在线程序http://hansunysb.edu/volume_estimation去估算体积。用户也可以手动输入体积。

(4) 如果你研究压力下的分子晶体，有时你可能需要稍微增加初始体积，为了能够用随机对称算子产生结构。

▷ *variable* `splitInto`

Meaning: 定义了单胞中的子晶胞或伪子晶胞数。如果你不想使用分裂的方法，就用数值1，或者删掉该块体。分裂选项仅适用于单胞中原子数大于25–30的情况。

Default: 1

Format:

```
% splitInto (number of subcells into which the unit cell is split)
1 2 4
% EndSplitInto
```

子晶胞采用了其他的平移对称性（赝）。除此之外，每个子晶胞可以使用一种特别的算法通过空间群构建，这个算法由A.R. Oganov 和H.T.Stokes开发，且由H.T. Stokes执行的特殊算法（见参考文献¹³）。

4.7 重新开始

如果某些作业运行出错，你想要在它停止的点重新开始计算—或者是更早的点。如果你想做的就是从停止的那一点处重新开始运行，你不需要改变任何设置（所有信息都会被存储在*.mat文件中），删掉still_reading文件，再次运行USPEX就足够了。

如果你想要在一个特殊结果results文件夹中的特殊的一代重新开始，那么给出pickUpGen = 你想要重新开始的代数。pickUpFolder = number of results文件夹的编号（例如results1为1, results2, ...）也是从你想要重新开始的那里开始。如果pickUpGen=0就是要重新开始计算。三个参数的默认值都为0。例如，想要从第10代开始在results5文件夹中重新启动运算，需设置：

```
10 : pickUpGen
5 : pickUpFolder
```

4.8 详述从头算法

USPEX采用强大的两级并行框架，这使得它成为并行扩展的典范。并行第一级是在结构弛豫代码里执行的，使每个结构可以达到约10-102个CPU的高效率。并行第二级是把计算分配给了统一群中的个体（因为同一种群内的结构之间是彼此独立的）。

首先，你需要指定你想使用哪个代码去进行结构弛豫和目标函数计算：

▷ *variable* `abinitioCode`

Meaning: Defines the code used for every optimization step.

Default: 1 for every optimization step (VASP)

Format:

```
% abinitioCode
3 2 2 1 1
% ENDabinit
```

Note:定义应用于每一步优化的准则。(所支持的代码详见章节 ??):

1 — VASP	7 — CP2K
2 — SIESTA	8 — Quantum Espresso
3 — GULP	9 — FHI-aims
4 — LAMMPS	10 — ATK
5 — Neural Networks code (unused at the moment)	11 — CASTEP
6 — DMACRYS	12 — Tinker
	13 — MOPAC

▷ *variable* `KresolStart`

Meaning: 明确而给出 k 点网格倒易空间的分辨率(单位: $2\pi\text{\AA}^{-1}$).

Default: 线性的从0.2到0.08

Format:

```
% KresolStart
0.2 0.16 0.12 0.08
% Kresolend
```

注意:你可以输入几个数值(每一步结构弛豫对应一个数),开始时粗略些(也就是值大一些),到最后,精确度高,这样就明显地加快了计算速度。特别是对金属而言,因为需要很多 k 点。如果你想用VASP或QuantumEspresso,那么这个模块非常重要(GULP完全没必要用它,而对于SIESTA,你必须在SIESTA输入文件里定义KresolStart)。

For **clusters**, **2D-crystals**, and **surfaces**,对于团簇, 2D-晶体和表面,你必须要明确给定团簇(或表面)真空范围的厚度。:

▷ *variable* `vacuumSize`

Meaning: 定义在结构周围增加的空间的数量(在毗邻的单胞内邻近团簇之间的最近邻距离, \AA)。仅适用于表面, 2D-晶体和纳米微粒。

Default: 10 每一步弛豫设为 10\AA

Format:

```
% vacuumSize
10 10 15 20 20
% EndVacuumSize
```

▷ *variable* `numParallelCalcs`

Meaning: 指出在并行计算中结构弛豫的数目。

Default: 1

Format:

```
10 : numParallelCalcs
```

你需要提供提交作业文件或者你正在使用的准则/模式下的可执行文件的名字。

▷ *variable* `commandExecutable`

Meaning: 明确给定提交作业文件名或在给定准则下的可执行文件名。

Default: 无默认值，需要用户自己设定。

Format:

```
% commandExecutable
gulp < input > output
mpirun -np 8 vasp > out
mpirun -np 8 vasp > out
mpirun -np 8 vasp > out
% EndExecutable
```

注意: 每一行都对应于一个弛豫阶段——第一行描述了第一个弛豫阶段的执行，等等。例如：从头算准则等于“3 1 1 1”意味着第一个弛豫阶段是执行GULP，接下来的阶段会通过VASP用命令“mpirun -np 8 vasp > out”用VASP命令执行。如果在可执行命令中只显示出一行，那么所有的弛豫阶段都执行相同的命令。

实际上在你远程提交作业模式下几乎所有平台都可以使用USPEX，你所需要的是在你的工作站有MATLAB/Octave运行。在这种情况下，你的工作站准备开始输入（包括jobs），然后提交到远程计算节点上，计算完成时查看结果并分析，准备新的输入文件。传出和来往的数据不是很大，所以网络不需要非常快。当然，作业提交依赖于机器。

▷ *variable* `whichCluster`

Meaning: 明确给定作业提交的类型。

Possible values (integer):

- 0 — 无作业脚本；
- 1 — 本地提交；
- 2 — 远程提交。

Default: 0

Format:

```
1 : whichCluster
```

▷ variable `remoteFolder`

Meaning: 计算即将进行的远程超级计算机上的文件夹。只有当`whichCluster=2`时，这个文件夹才会被激活。

Default: 没有

Format:

```
Blind_test : remoteFolder
```

注意: 在远程提交文件夹中有一个相似的参数设置— `homeFolder`，计算的真实路径实际上是`*homeFolder*/*remoteFolder*/CalcFolderX`，在此处，`X=1, 2, 3,...`。

▷ variable `PhaseDiagram`

Meaning: 使计算类型=300 和301的相图的计算能够进行。这给出了一个想法（比较粗劣，仅仅是一个大致的结果）：给出新相可能会在较高或较低压力下，同时粗劣的给出一个转变压力。

Default: 0

Format:

```
1 : PhaseDiagram
```

4.9 指纹设置

在改变下面这些参量之前，建议你先读一下关于指纹概念和方法论的文章，这样你才能知道你在设置的是什么（Oganov & Valle, 2009¹⁸）。

▷ variable `RmaxFing`

Meaning: 截断距离(单位Å)。

Default: 10.0

Format:

```
10.0 : RmaxFing
```

▷ variable `deltaFing`

Meaning: 是指纹函数的离散化(单位Å)。

Default: 0.08

Format:

```
0.10 : deltaFing
```

▷ variable `sigmaFing`

Meaning: 是原子间距的高斯扩展。

Default: 0.03

Format:

0.05 : sigmaFing

`toleranceFing`（默认值为0.003）和`toleranceBestHM`（默认值为0.02）指定了结构之间的最小余弦距离，证明它们是不相同的——分别用于产生子代和适者生存。它们依赖于结构弛豫的精度和体系的物理性质（例如对于排序问题，分别属于不同结构的指纹非常相似，并且参数公差也设得非常小）只有当`dynamicalBestHM=1`时（我们已不用了），`toleranceBestHM`才是有用的，所以这个参数基本上被废弃。

4.10 反种子设置

反种子技术已经在USPEX中开发实施了，所有基于对结构不利的想法都已被屏蔽以保证模拟没有陷入局部最小值。在这儿，和时间有关的适应度是所有总的真实焓值（或其他的适应性质）和跟历史有关的项，它是能量抽样面上叠加的高斯势能之和：

$$f = f_0 + \sum_a W_a \exp\left(-\frac{d_{ia}^2}{2\sigma_a^2}\right),$$

在这里， f 是适应度（ f_0 — 真实适应度， f — 跟历史有关的适应度）， W_a 是高度，是高斯展宽。在我们的方法中，高斯参数的变化依赖于种群多样性，能量在每一代中扩展。

有三种使用该技术的方法。第一种，将你认为不利的结构放置在反种子文件夹**AntiSeeds**中。例如它可能是基态结构—在这种情况下，USPEX将会尽力找到第二个最低焓值的结构。

在第二种和第三种方法中，你不需要明确给定反种子结构—USPEX要么用所有的抽样结构作为反种子（经过很好的测试，是被推荐的方法），要么是每一代中的最优结构。你需要给定以下的一些设置：

▷ *variable* `antiSeedsActivation`

Meaning: 明确从哪一代开始反种子模式开始。当`antiSeedsActivation = N > 0`时，从第 N 代开始，高斯势加入到所有的结构中，当 $N < 0$ 时，高斯势仅加入到每一代的最优结构中，从第 N 代开始。当 $N = 0$ 时，高斯势仅加入到放在反种子文件夹中的结构中。如果你不想用反种子，那么给定非常大的`antiSeedsActivation`值（例如:5000）和`antiSeedsMax=0.0`。

Default: 5000

Format:

1 : antiSeedsActivation

▷ variable `antiSeedsMax`

Meaning: 明确高斯的高度，以每一代中的焓值的平均方差为单位（算出来的值仅在`bestFrac`之间，例如在准双亲之间）。我们推荐`antiSeedsMax=0.01`。

Default: 0.000

Format:

```
0.005 : antiSeedsMax
```

▷ variable `antiSeedsSigma`

Meaning: 明确高斯的宽度，以每一代中结构之间的平均距离为单位（算出来的值仅在`bestFrac`之间，例如在准双亲之间），我们建议`antiSeedsSigma=0.005`。

Default: 0.001

Format:

```
0.005 : antiSeedsSigma
```

Fig. 7 是一个应用反种子技术的例子。

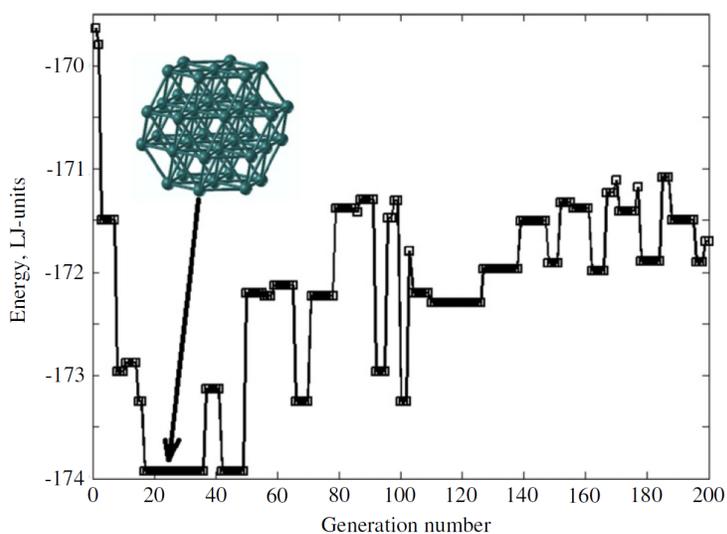


Figure 7: 应用反种子的有38个原子的Lennard-Jones集群计算的例子。每一代中的最优结构的能量已被标绘出。我们可以清晰地看到，算法没有陷在任何一个候选最小值很长时间，并且很快发现了基态。在这儿我们用`antiSeedsActivation=1`, `antiSeedsMax=0.01`, `antiSeedsSigma=0.001`。

4.11 空间群的判定

▷ variable `doSpaceGroup`

Meaning: 判定空间群，还以结晶学的*.CIF格式（当你要准备发表的时候这会更简单一点，但是需要注意的是：如果弛豫不是非常精确，或者是如果对于对称探测器设置了非常严格的公差，那么有时候空间群可能是不确定的）写在输出文件中。多亏了由H.T. Stokes提供的强大的对称准则，这个选项才得以被使用。

Default: 1, if calculationType=3** (300, 301, 310,311 — bulk crystals) and 0 otherwise.

Format:

1 : doSpaceGroup (0 - 没有空间群, 1 - 判定空间群)

▷ variable **SymTolerance**

Meaning: 用H.T. Stokes的寻找对称性代码精确地确定对称性。可以设置为一个数字（单位为Å），或者是high | medium | low (= 0.04 | 0.08 | 0.15)。

Default: medium

Format:

medium : SymTolerance

4.12 给开发者的关键词

▷ variable **repeatForStatistics**

Meaning: USPEX自动运行的数量。USPEX模拟是随机的，用同样的输入参数再次模拟，得到的结果是必然不同的。然而最终的结果—基态值—是相同的（也是我们所期望的）。为了达到它所运行的步数和化学空间中的轨道数在每次运行时也是不同的。为了比较不同的算法，你必须收集一些统计资料—不要只依赖于单次运行（它可能是好运的，也可能不是好运的...USPEX不能只依赖于好运）。这个选项仅是开发者感兴趣，而且仅对用简单的势函数（如GULP）收集统计资料有意义。

Default: 1 (i.e., 也就是说，没有统计资料将会被收集)

Format:

20 : repeatForStatistics

▷ variable **stopFitness**

Meaning: 设置适应度的值就可以当达到适应度值 \leq stopFitness时，计算停止。

Default: 没有默认值，需要用户自己设定。

Format:

90.912 : stopFitness

注意: 当stopFitness值给定之后，统计资料才能够自动分析。为了收集stopFitness的

可获取的统计资料建议repeatForStatistics模块部分设置的值>。输出样本如下:

```

Number of files to be processed: 20
Target enthalpy: 90.912

Generation: 23   Number: 1326   Enthalpy: 90.9119   Mat-file: /home/USPEX/01/results1/USPEX.mat
Generation: 22   Number: 1224   Enthalpy: 90.9119   Mat-file: /home/USPEX/02/results1/USPEX.mat
Generation: 60   Number: 3451   Enthalpy: 90.9119   Mat-file: /home/USPEX/03/results1/USPEX.mat
Generation: 30   Number: 1739   Enthalpy: 90.9119   Mat-file: /home/USPEX/04/results1/USPEX.mat
Generation: 17   Number: 956    Enthalpy: 90.9119   Mat-file: /home/USPEX/05/results1/USPEX.mat
Generation: 36   Number: 2055   Enthalpy: 90.9119   Mat-file: /home/USPEX/06/results1/USPEX.mat
Generation: 35   Number: 1987   Enthalpy: 90.9119   Mat-file: /home/USPEX/07/results1/USPEX.mat
Generation: 22   Number: 1241   Enthalpy: 90.9119   Mat-file: /home/USPEX/08/results1/USPEX.mat
Generation: 18   Number: 1002   Enthalpy: 90.9119   Mat-file: /home/USPEX/09/results1/USPEX.mat
Generation: 29   Number: 1641   Enthalpy: 90.9119   Mat-file: /home/USPEX/10/results1/USPEX.mat
Generation: 21   Number: 1197   Enthalpy: 90.9119   Mat-file: /home/USPEX/11/results1/USPEX.mat
Generation: 27   Number: 1542   Enthalpy: 90.9119   Mat-file: /home/USPEX/12/results1/USPEX.mat
Generation: 44   Number: 2519   Enthalpy: 90.9119   Mat-file: /home/USPEX/13/results1/USPEX.mat
Generation: 32   Number: 1821   Enthalpy: 90.9119   Mat-file: /home/USPEX/14/results1/USPEX.mat
Generation: 15   Number: 835    Enthalpy: 90.9119   Mat-file: /home/USPEX/15/results1/USPEX.mat
Generation: 43   Number: 2477   Enthalpy: 90.9119   Mat-file: /home/USPEX/16/results1/USPEX.mat
Generation: 40   Number: 2278   Enthalpy: 90.9119   Mat-file: /home/USPEX/17/results1/USPEX.mat
Generation: 24   Number: 1358   Enthalpy: 90.9119   Mat-file: /home/USPEX/18/results1/USPEX.mat
Generation: 14   Number: 757    Enthalpy: 90.9119   Mat-file: /home/USPEX/19/results1/USPEX.mat
Generation: 27   Number: 1532   Enthalpy: 90.9119   Mat-file: /home/USPEX/20/results1/USPEX.mat

Found structures numbers : 1326 1224 3451 1739 956 2055 1987 1241 1002 1641 1197 1542 2519 1821 835 2477 2278 1358 757 1532
Found generations numbers: 23 22 60 30 17 36 35 22 18 29 21 27 44 32 15 43 40 24 14 27

Success rate: 100 percent
Average number of generations to get E=90.912: 29
Average number of structures to get E=90.912: 1647
Standard deviation: 670

```

▷ *variable* `collectForces`

Meaning: 收集USPEX计算产生的所有弛豫优化的信息，包括在结构优化中的原子上的力，原子位置，晶格常数和应力张量。这个信息存储在FORCE.mat文件中。仅VASP支持此功能。

Default: 0

Format:

1 : collectForces

4.13 不常使用的关键字

▷ *variable* `ordering_active`

Meaning: 通过局部有序参数打开有偏见的变量算符。

Default: 1

Format:

1 : ordering_active

▷ *variable* `symmetrize`

Meaning: 把所有结构转换为可以适应对称性的结晶学设置。

Default: 0

Format:

```
1 : symmetrize
```

▷ *variable* `valenceElectr`

Meaning: 每种原子类型的价电子数。

Default: 所有原子的值都是常量，我们已经将它们制成了表，不需要特别设置。

Format:

```
% valenceElectr
2 6
% EndValenceElectr
```

▷ *variable* `percSliceShift`

Meaning: 在所有维度内进行块移动（在遗传中用到）的概率，1.0 代表100%。

Default: 1.0

Format:

```
0.5 : percSliceShift
```

▷ *variable* `dynamicalBestHM`

Meaning: 当`keepBestHM`为上限值时，设置在计算中存活的最优结构的数量是否不同。
可能的值（整数）：0=没有突变；1和2=看注解

Default: 2

Format:

```
1 : dynamicalBestHM
```

注意: 如果你设置`dynamicalBestHM=1`，那么代码就会选择`keepBestHM`的最低能量结构（没有重复结构，这可以定义为有低于用户定义的`toleranceBestHM`的指纹距离）。如果`dynamicalBestHM=2`（我们的优先选择），在与`bestFrac`相一致的全部能量间隔内，聚类算法恰好选择了`keepBestHM`的最大不同的结构，并且最佳`toleranceBestHM`值是自动决定的——在保留最优结构记忆的同时促进了多样性的发展。

▷ *variable* `softMutOnly`

Meaning: 仅由软模变异可以产生多少代。

Default: 0

Format:

```
% softMutOnly
1-5
```

```
% EndSoftOnly
```

注意: 在上面的例子中, 仅由软模变异产生的代数可以达到 5^{th} 代 (当然, 包括第一代)。注意根据软模变异, 每个亲代产生两个软模变异突变体。你也可以在运行的过程中给定经过软模变异的特殊的几代, 例如, 每 10^{th} 代软模变异一次, 你可以这样写:

```
% softMutOnly
2 12 22 32 42
% EndSoftOnly
```

▷ *variable* `maxDistHeredity`

Meaning: 给定参与遗传的结构之间的最大余弦距离。它指定了与该结构能够匹配的能量面半径。

Default: 0.5

Format:

```
0.5 : maxDistHeredity
```

▷ *variable* `manyParents`

Meaning: 明确是否应该要超过两片 (或超过两个亲代结构) 被用于遗传。这对大体系来说是有用的。

可能的值(整数):

0 — 仅用两个亲代, 每个一个片段。

1 — 许多结构被用作亲代, 每个一个片段。

2 — 两个结构被用于亲代, 许多片段 (用参数`minSlice`和`maxSlice`动态决定) 可以从每个亲代中独立选择。

3 — 两个结构被用于亲代, 许多片段 (用参数`minSlice`和`maxSlice`不断变化决定) 是从有特定设置的晶胞中切下来。这是适用于大体系的较好的选择。例如: 我们将两个结构切成近似相等厚度的薄片, 然后从亲代1中选择偶数片, 从亲代2中选择奇数片, 制成多层的“三明”或是“斑马”。

Default: 0

Format:

```
3 : manyParents
```

`minSlice`, `maxSlice`: 决定了切片的最小和最大厚度, 单位为 \AA , 他们将会从亲代中删掉来参加子代结构的产生。我们期望这些薄片足够薄, 能够携带一些亲代的信息 (但不是太薄, 以至于对遗传没有影响)。这些参数的合适值分别大约是1和6 \AA 。

对于团簇, 你可以直接给定参与遗传的亲代数 (但是我们发现这几乎没有用):

▷ *variable* `numberparents`

Meaning: 定义了团簇中遗传的亲代数。

Default: 2

Format:

2 : numberparents

5 特殊情况的额外输入

5.1 分子晶体的MOL_1, MOL_2, ... 文件

5.1.1 分子晶体, calculationType=310/311

对于一个分子晶体, MOL_1文件描述分子结构开始构建的结构部分。这个文件也定义如果分子可活动的哪个扭转角会突变。这个文件和它的格式跟SIESTA's *Z_Matrix*文件 (MOL_1给出原子的笛卡尔坐标, 反之, *Z_Matrix*文件从键长、键角和扭转角定义原子位置) 不同。*Z_Matrix*文件是利用MOL_1文件给的信息创建的, 即: 键长和所有必要的角度都是用笛卡尔坐标计算的。重要的键长和角度用来创造*Z_Matrix*—这恰好是5-7列列举的。我们看一下苯C₆H₆的MOL_1文件:

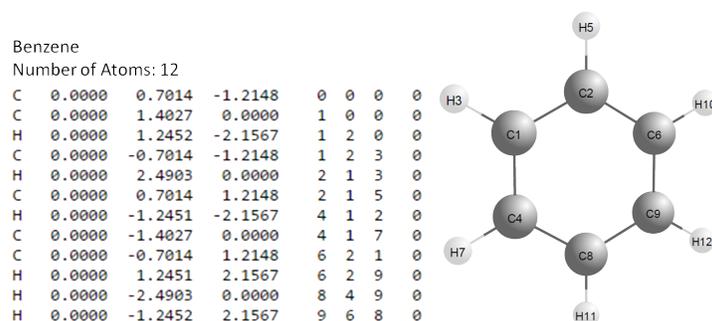


Figure 8: 苯的MOL_1示例和其相应的分子结构示意图。

第一个原子是H, 它的坐标定义与其他原子无关 (“0 0 0”).

第二个原子是C, 它在*Z_matrix*的坐标 (在分子坐标系中) 仅根据它到第一个原子 (即, 上面描述的H) 的距离来设置, 但没有角度 (“1 0 0”).

第三个原子是C, 它的坐标将根据它到第二个原子的距离来设置, 键角3-2-1, 但不是扭转角—因此我们用 “2 1 0”。

第四个原子是C, 它的坐标根据它到第三个原子的距离设置, 键角4-3-2, 扭转角4-3-2-1—因此, 我们用 “3 2 1” 等等... 知道最后, 第12个原子, H, 根据它到第七个原子 (C) 的距离定义, 键角12-7-6, 扭转角12-7-6-11—因此是 “7-6-11”。

最后一列是扭转角灵活度的标示。例如, 在C4中, 扭转角由4-3-2-1定义。理想情况下, 对于前3个原子来说这个标示应该是1, 其他原子的是0, 如果存在其他活动的扭转角, 需在这一列说明是1。

5.1.2 聚合物晶体, calculationType=110 (“线性链状模型”)

对于聚合物, MOL_1文件用来表示单体单元的几何结构, 和分子晶体一样, 除了要在最后一行列出活性原子, 如下图列出的PVDF (聚偏二氟乙烯) 的MOL_1文件:

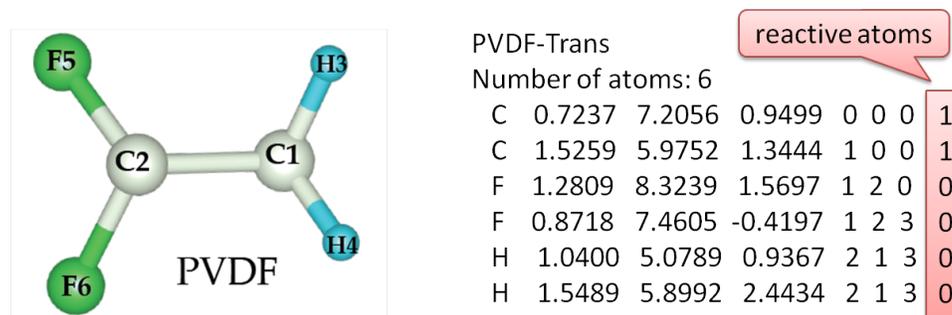


Figure 9: PVDF的MOL_1示例和相应的单体结构。

5.1.3 经典力场的额外输入

上面的MOL_1文件适用于USPEX的一般情况。然而，一些基于编码的经典力场需要额外的信息。例如，GULP需要列出化学分类和电荷。阿司匹林的MOL_1文件的写法如下：

```
Aspirin_charge
Number of atoms: 21
H_1      0.2310      3.5173      4.8778      0 0 0 1      0.412884
O_R      0.7821      4.3219      4.9649      1 0 0 1     -0.676228
C_R      0.4427      5.0883      6.0081      2 1 0 1      0.558537
O_2     -0.5272      4.5691      6.6020      3 2 1 0     -0.658770
C_R      1.0228      6.3146      6.3896      3 2 4 0      0.116677
C_R      2.1330      6.8588      5.6931      5 3 2 0      0.311483
C_R      0.4810      7.0546      7.4740      5 3 6 0     -0.119320
O_R      2.8023      6.2292      4.6938      6 5 3 0     -0.574557
C_R      2.6211      8.1356      6.0277      6 5 8 0     -0.083091
C_R      0.9966      8.3146      7.8237      7 5 3 0     -0.103442
H_2     -0.3083      6.6848      8.0128      7 5 10 0     0.198534
C_R      3.6352      5.1872      4.9079      8 6 5 0      0.609295
C_R      2.0623      8.8613      7.0940      9 6 5 0     -0.119297
H_2      3.3963      8.5283      5.4906      9 6 13 0     0.174332
H_2      0.5866      8.8412      8.6013     10 7 13 0     0.205960
O_2      3.9094      4.7941      6.0632     12 8 6 0     -0.588433
C_3      4.2281      4.5327      3.7638     12 8 16 0    -0.271542
H_2      2.4227      9.7890      7.3367     13 9 10 0     0.196738
H_2      3.4269      4.1906      3.1183     17 12 8 0     0.151315
H_2      4.8283      3.6848      4.0792     17 12 19 0    0.131198
H_2      4.8498      5.2464      3.2337     17 12 19 0    0.127726
```

这里，标题的关键词电荷是告诉程序在最后一行读取电荷。

结合Tinker一起计算，附加行用来列出原子标签，如下所示：

Urea

Number of atoms: 8

C	0.000000	0.000000	0.000000	0 0 0 1	189
O	0.000000	0.000000	1.214915	1 0 0 1	190
N	1.137403	0.000000	-0.685090	1 2 0 1	191
N	-1.137403	0.000000	-0.685090	1 2 3 0	191
H	1.194247	0.000000	-1.683663	4 1 3 0	192
H	-1.194247	0.000000	-1.683663	4 1 3 0	192
H	1.998063	0.000000	-0.138116	2 1 3 0	192
H	-1.998063	0.000000	-0.138116	2 1 3 0	192

5.1.4 如何准备MOL文件

有许多程序可以生成Zmatrix style文件，如Molden、Avogadro等。有经验的用户或许有自己的方法来准备这些文件。为了用户的方便，我们创建了一个在线工具，允许用户仅用XYZ格式的文件来生成USPEX-style MOL文件。这个工具的网址为<http://han.ess.sunysb.edu/zmatrix>。

5.2 表面

为了完成表面重构的预测，必须：

- 指定200 或201 : 计算类型。
- 准备包含基底的VASP5 POSCAR格式的文件，如图11 10所示。
- 列举以下参数：

```
% symmetries
2-17
% endSymmetries
```

注意：如果对称标签symmetries出现，USPEX将尝试使用平面组生成结构。

▷ *variable* **thicknessS**

Meaning: 表面区域的厚度。只有这个区域允许吸附原子。

Default: 2.0 Å

Format:

```
3.5 : thicknessS
```

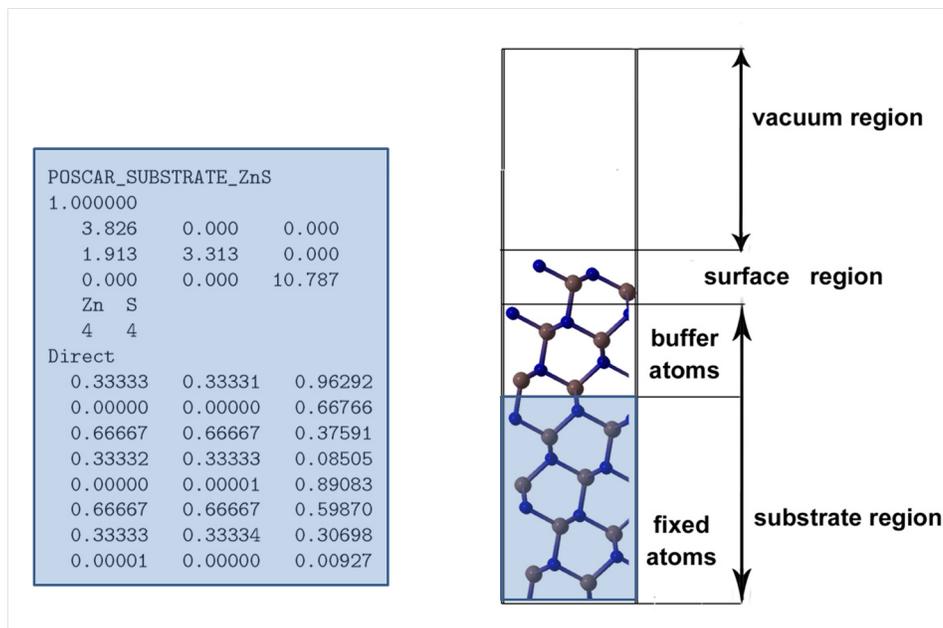


Figure 10: USPEX使用的表面模型。注意POSCAR_SUBSTRATE 应该准确代表没有真空的晶体的几何信息。如果输入中有大量的真空区域，程序将自动删除它并提供一个名为POSCAR_SUBSTRATE_NEW的新文件，这个文件必须在计算中用到（重命名为POSCAR_SUBSTRATE）。

▷ variable **thicknessB**

Meaning: 基底的缓冲区域的厚度。这个区域是POSCAR_SUBSTRATE的一部分，可以松弛。

Default: 3.0 Å

Format:

3.0 : thicknessB

▷ variable **reconstruct**

Meaning: 表面单元的最大增值，允许复杂重建。

Default: 1

Format:

1 : reconstruct

USPEX也提供研究表面原子变成分搜索的可能（在USPEX 10.1发布）。在这种情况下，稳定的表面重建是由化学势²⁵控制的。典型的输入设置如下：

```
*****
*      TYPE OF RUN AND SYSTEM      *
*****
```

```
USPEX : calculationMethod (USPEX, VCNEB, META)
201   : calculationType (dimension: 0-3; molecule: 0/1; varcomp: 0/1)
```

```
% atomType
Si 0
% EndAtomType
```

```
% numSpecies
2 4
% EndNumSpecies
```

这里我们列出 1×1 晶胞表面原子的最大数。

```
*****
*                SURFACES                *
*****
% symmetries
2-17
% endSymmetries

% StoichiometryStart
1 2
% StoichiometryEnd
```

这定义晶体的化学计量数。

```
-23.7563 : E_AB (AmBn化合物的DFT能量, eV/化学式)
-5.4254  : Mu_A (元素A的DFT能量, eV/原子)
-4.9300  : Mu_B (元素B的DFT能量, in eV/atom)

3.5      : thicknessS (表面区域的厚度, 默认2  $\text{\AA}$ )
3.0      : thicknessB (基底缓冲区域的厚度, 默认3  $\text{\AA}$ )
4        : reconstruct (晶胞的最大倍数)
```

现在, USPEX支持如下的表面变成分计算:

- 基本的表面重建 (如C在diamond(100)表面)。
- 二元化合物的表面重建 (如GaN(0001)表面)。
- 包含不同物种的基本表面的重建 (如PdO在Pd(100)表面)。

5.3 变成分代码

切换为变成分模式，必须：

1. 设置301、311或201 : 计算类型。
2. 在numSpecies下设置结构组成(参照numSpecies的说明书)。
3. 用Latticevalues设置每种类型原子（或每个组成块）粗略的原子体积。
4. 设置下面的varcomp-only选项：

▷ variable **firstGeneMax**

Meaning: 第一代抽取多少不同的组分。如果是0，这个数值就等于initialPopSize/4。对于二元的，要求firstGeneMax=11，对于三元，需要更高的值，比如30。

Default: 11

Format:

```
10 : firstGeneMax
```

▷ variable **minAt**

Meaning: 第一代单胞的最小原子数（对于calculationType=301/201/300）或分子数(对于calculationType=311)。

Default: 无默认值

Format:

```
10 : minAt
```

▷ variable **maxAt**

Meaning: 第一代单胞的最大原子数（对于calculationType=301/201/300或META计算）或分子数（对于calculationType=311）。

Default: No default

Format:

```
20 : maxAt
```

▷ variable **fracTrans**

Meaning: 通过原子替换获得的结构所占的百分比。这个过程中，一个随机选取的原子被系统中其他化学成分的原子取代—新的化学组成由默认随机选取，或者你可以在specificTrans下定义它，就像明确的置换。

Default: 0.1

Format:

```
0.1 : fracTrans
```

▷ *variable* `howManyTrans`

Meaning: 结构中被替换原子的最大百分比 (0.1 = 10%)。要替换的原子的百分比从0到`howManyTrans`随机选取。

Default: 0.2

Format:

```
0.2 : howManyTrans
```

▷ *variable* `specificTrans`

Meaning: 允许替换的种类。

Default: 空行 (没有特定的替换)

Format:

```
% specificTrans
1 2
% EndTransSpecific
```

注意: 在这种情况下, 原子1可以被转变成原子2, 反之原子2也可以被转变成原子1. 如果你想尝试所有的变形可能, 在这一关键块下留下空行。

变成分案例在运行时, 参数`keepBestHM`有新含义—凸包 (比如, 多成分系统的热力学稳定状态) 图上的所有结构保留, 连同convex hull附近的一些亚稳结构—总量是`keepBestHM`。

对于变成分的运行, 明智地设置第一代尤其重要。选择适当大的初始代的大小`initialPopSize`。为不同组成的`firstGeneMax`选取合理大的数值作为第一代的样品。最后, `minAt`和`maxAt`的差异不超过2倍, 而且不同的体系大小你可能需要少量的计算: 比如, 4-8, 8-16, 16-30原子, 等等。

给VASP用户一个附加意见—如果你想完成一个变成分的运行, 拿Na-Cl体系来说, 你需要确定`INPUT.txt`中给出了正确的原子类型, 并且将赝势文件`POTCAR_Na`和`POTCAR_Cl`放入文件夹`~/StructurePrediction/Specific/`。USPEX将识别每个原子并且适当选取每个原子的`POTCAR`文件进行计算。图Fig. 11展示了USPEX发现的且被实验证实²⁶的热力学稳定的氯化钠。

5.4 进化准动力学命令

这对于找全局最小值是一个非常强大的方法, 以及开始结构中潜在的动力学接近的低能量亚稳结构。开始的结构必须是高质量的且在`POSCAR_1`中给出。进化准动力学只能

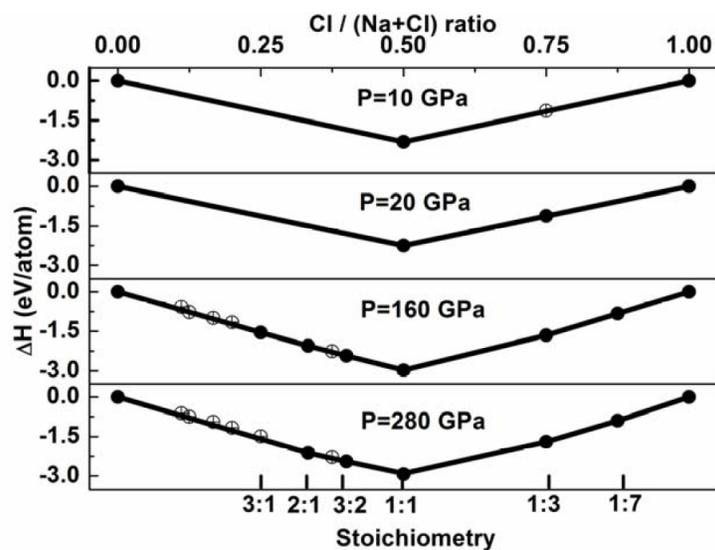


Figure 11: 在不同压力下Na-Cl体系的凸包图。实心圆圈代表稳定的结构；空心圆圈代表亚稳定的结构。

在VASP和GULP中使用。

为了运行进化准动力学程序，必须：

1. 设置

`META` : 计算方法

`300` : 计算类型

2. 在你的文件夹的VASP5版本中建立POSCAR_1文件（进化准动力学需要一个好的初始结构，在感兴趣的压力下弛豫）。

3. 设定数量大小（这个例子中，这是每一步软模变异的数量）：

`30` : `populationSize`

4. 设定压力:

▷ *variable* `ExternalPressure`

Meaning: 你想在哪个压力下执行计算，单位GPa.

Default: 无默认值

Format:

`10` : `ExternalPressure` (GPa)

5. 设置准动力学选项:

▷ variable **GaussianWidth**

Meaning: 每个加入到能量面以加速相变的高斯宽度。好的经验是从 $0.10-0.15L$ 选一个值，这里 L 是单胞的最小长度，单位为Angstroms。

Default: $0.10 \times L$ (Å)

Format:

0.80 : GaussianWidth

▷ variable **GaussianHeight**

Meaning: 每个加入到能量面以加速相变的高斯高度。好的经验(Marotoňák *et al.*, 2005)是选一个接近 $L(\delta h)^2 G$ ， L 是单胞的平均长度，单位为Angstroms， δh 是高斯宽度， G 是剪切模量，单位为kbars。

Default: $1000 \times (0.10 \times L)^2 \times L = 10 \times L^3$ (Å³ kbar)

Format:

2000 : GaussianHeight

▷ variable **FullRelax**

Meaning: 这样的准动力学仅在固定晶胞中弛豫结构。为了分析，我们需要进行完整的结构弛豫（即弛豫晶胞）。

- **FullRelax=0** — 没有完全弛豫要执行（非常快速的选择，但对结果分析造成不便）。
- **FullRelax=1** — 仅一代中的最好结构要完全弛豫也很快，有时充足）。
- **FullRelax=2** — 所有的等价结构完全弛豫（依旧很快，只比**FullRelax=1**时慢~2倍，但提供了更多的内涵信息。大多数案例都需要它）。

Default: 2

Format:

2 : FullRelax

为了完全弛豫，当执行进化准动力学计算时，`abinitioCode`的格式略有不同，比如：

```
abinitioCode
3 3 3 3 (3 3)
ENDabinit
```

在上面的例子中，一个固定晶胞的弛豫有4个阶段，完全弛豫有2个（括号中）。记住，在最后的固定晶胞的弛豫阶段，压力张量必须精确—这才能驱动准动力学过程。目前只有VASP, SIESTA和GULP 支持此计算。

▷ variable `maxVectorLength`

Meaning: 与`minVectorLength`一样，是基础晶胞在进化准动力学模拟时的长度边界值（注意在与一般计算中`minVectorLength`含义不同，`maxVectorLength`仅在进化准动力学中使用）。当任何基础晶胞的长度比`minVectorLength`或比`maxVectorLength`大，我们会在准动力学中添加一个夸张的修正力，这驱使晶胞进化朝“好”的方向发展。当所有基础晶胞的长度在“好的”范围时，这个修正力为零。

Default: 无默认值

Format:

12.0 : `minVectorLength`

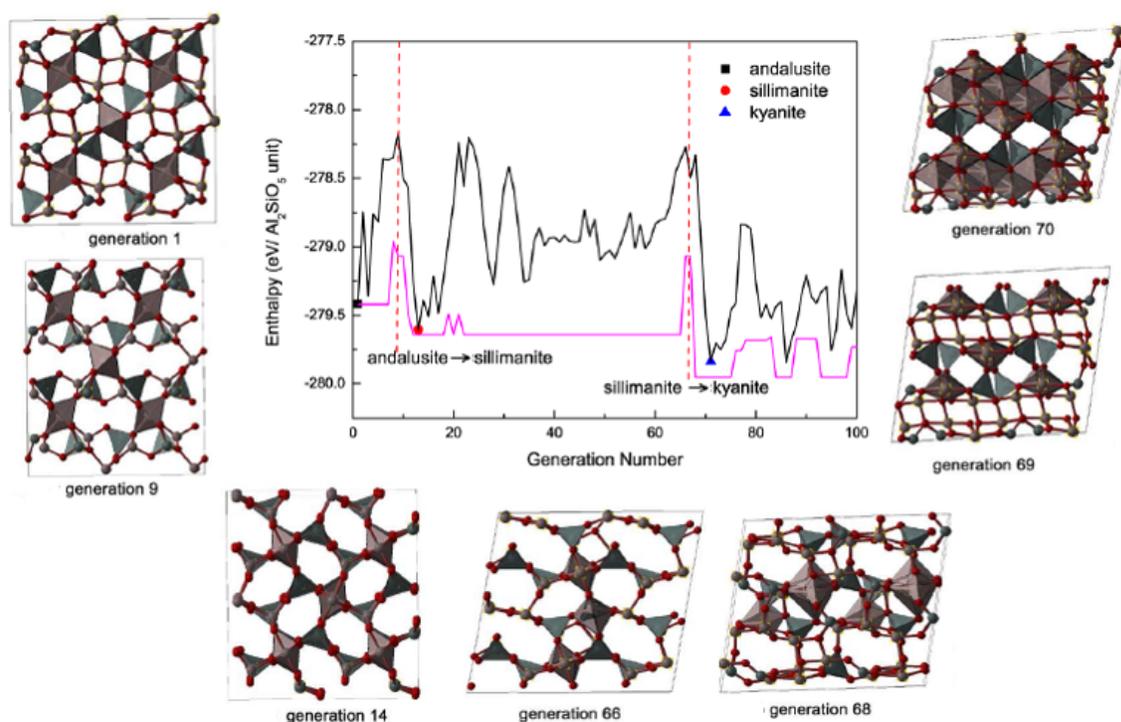


Figure 12: 10GPa时压缩红柱石 (Al_2SiO_5) 过程中的焓演变（黑线：常量 h 下最好结构的焓；红线：最好的结构完全弛豫后的焓）。得到结构的顺序为：第1代（红柱石）→第9代（硅线石）→第14代→第66代→第68代→第69代→第70代（蓝晶石）。

当运行准动力学时，附加文件可在`results1`文件夹中找到，更重要的是：

- `force.dat` — 分析晶胞上的力，内部的(`f_c`)和来自高斯的 (`f_g`)；
- `presten` — 压力张量；
- `lattice.dat` — 模拟过程中晶胞形状的改变；

- `enthalpies` and `enthalpies_relaxed` — 固定晶胞每一步结构的焓和完全弛豫后的焓;
- `gatheredPOSCARS` and `gatheredPOSCARS_relaxed` — 固定晶胞的结构和完全弛豫后的结构。

图Fig. 12展示了进化准动力学应用的例子：从一个 Al_2SiO_5 同质多形体（红柱石）开始，我们得到另外两个已知的同质多形体（蓝晶石和硅线石）和重要的相变机理。

5.5 粒子群算法（PSO）代码

在晶体和簇结构预测领域，一些算法对于小体系非常成功。粒子群算法，在这个领域由Boldyrev²⁷所倡导，是进化算法的特殊情形，大量的（集群）候选值（称为“微粒”）根据一些简单的公式移动到搜索空间。微粒的运动由它们在搜索空间中自己最显著的位置决定，和全体集群最显著的位置一样。首先，微粒的坐标 χ 和‘速度’ v 随机产生。随后每一步，位置和速度都根据以下公式更新：

$$\begin{aligned} v'_i &= \omega \cdot v_i + \varphi_p \cdot r_p \cdot (p_i - \chi_i) + \varphi_g \cdot r_g \cdot (g - \chi_i), \\ \chi'_i &= \chi_i + v'_i. \end{aligned} \quad (7)$$

这里 ω ， φ_p 和 φ_g 是加权因子，控制PSO算法的行为和效率； r_p 和 r_g 是[0; 1]范围内的随机数，分别在每一步为每一个微粒产生； p_i 是微粒 i 最显著的位置， g 是整个集群最显著的位置。

这种算法，尽管它简单但可以工作²⁷。改进之前安装运行^{27;28}的关键：（1）搜索空间的度量（单独在坐标系中绘制各个晶体结构不重要）；（2）用PSO进化结构的方法，也就是变异操作。

通过直接由原子坐标和两个结构的参数决定的速度 v_i (7)来进化微粒是不可行的。我们的问题是用指纹距离¹⁸作为能谱最原始的坐标，且USPEX用来进化PSO微粒的变量作为进化种群结构最有效公正的方法。即，微粒要么突变（模拟自由移动），要么是以最佳位置参与遗传或最佳的种群位置（模拟PSO朝这些方向移动）参与遗传。代替了所有的移动的每一步都加上一些权重(see Eq. 7)，我们通过以下描述的可能性一次性加上：

$$\begin{aligned} P_m &= \frac{\omega}{\Sigma}; & P_p &= \frac{\varphi_p \cdot r_p \cdot D_p}{\Sigma}; & P_g &= \frac{\varphi_g \cdot r_g \cdot D_g}{\Sigma}; \\ \Sigma &= \omega + \varphi_p \cdot r_p \cdot D_p + \varphi_g \cdot r_g \cdot D_g, \end{aligned} \quad (8)$$

其中 D_p 是一个微粒现在和最佳位置间的指纹间距， D_g 是微粒现在位置和全部种群最佳位置的指纹间距。我们的测试在几个不同的系统进行，表明这种方法（我们称之为“cor-PSO”，即校正PSO）是比较成功的。而且比PSO之前的版本好，但在成功率和效率方面仍不能和USPEX算法【3,41】相比。

下面的变量对`calculationMethod=PSO`是唯一的：

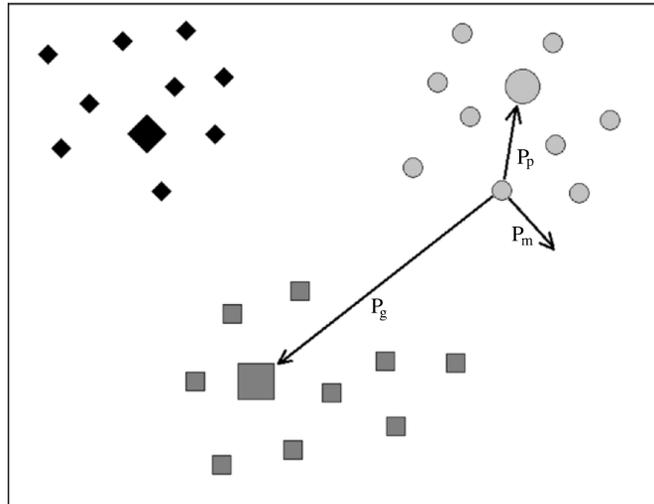


Figure 13: 三个个体数量10代之后的的PSO-USPEX混合算法图例（用菱形、正方形和圆形标记）。每个微粒的最好位置都用放大的符号标记。最好的结构是大的正方形。圆形标记的结构可突变，用其历史中最好的位置（大圆）产生子代或通过遗传操作以 P_m 、 P_p 和 P_g 的概率产生全部种群（大正方形）的最好位置。

▷ *variable* `PSO_softMut`

Meaning: 软模变异的权重（公式eq. 8中的 ω ）

Default: 1

Format:

1 : PSO_softMut

▷ *variable* `PSO_BestStruc`

Meaning: 用同样PSO微粒的最好位置遗传的权重(φ_p in eq. 8)。

Default: 1

Format:

1 : PSO_BestStruc

▷ *variable* `PSO_BestEver`

Meaning: 由全局遗传最好PSO的微粒(φ_g 的权重eq. 8)。

Default: 1

Format:

1 : PSO_BestEver

6 相变路径预测

相变决定材料许多方面的行为。因此，很有必要揭示结构相变可能存在的机理。

6.1 变胞微动弹性带（VCNEB）法

预测相变机制可以认为是一个两端的问题，算法必须定位在中间状态。变胞微动弹性带（NEB）^{29;30;31}方法是解决两端问题广泛运用的技术，一个有效而稳健的方法来寻找反应路径和鞍点，在“最小能量路径”（MEP）上的两个端点之间的势能面。该NEB方法已成功应用于分子化学反应和缺陷迁移，尤其它可以提供相变过程给定的初始和最终状态间的能量势垒。不幸的是，大多数用NEB方法解决的问题都是在单胞的限定下，这排除了它用于相变的可能（这涉及沿相变路径变异的单胞）。

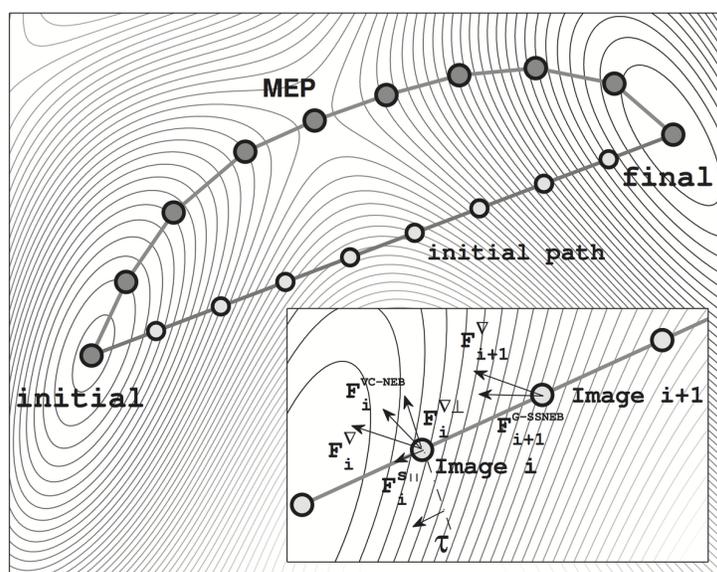


Figure 14: 2D無表面模型的最小能量路径（有灰色圆的线）和初始路径。影像*i*用VC-NEB方法的力示于插图。 F_i^∇ 是梯度方向的潜在力。 $F_i^{\nabla\perp}$ 和 $F_i^{\nabla\parallel}$ 分别是的横向分量和弹性力。

变胞NEB（VC-NEB）方法¹⁴，我们用不同的公式来完善，平等对待单胞和原子坐标，在恒定压力下在膨胀的结构空间运行。以第一性原理为框架的VC-NEB方法已经被添加到USPEX代码中。VC-NEB方法是探索更大结构空间相变过程两个端点激活路径的常用工具。VCNEB方法中路径上的所有结构都被示为“Image”。

6.2 VCNEB的Input选项

VCNEB方法目前只能用VASP，GULP和Quantum Espresso指令。

为了切换VCNEB模式，必须：

1. 设定

VCNEB : `calculationMethod`

2. 在VASP4版本你的文件夹中创建**Images**文件（VCNEB需要至少两个结构，初始和最终相，来进行相变路径预测）。

3. 设置以下VCNEB选项：

▷ variable `vcnebType`

Meaning: 说明VCNEB计算的类型。这个变量包含3个目录：*calculation option*, *Image number variability*, 和 *spring constant variability*:

- calculation option:
 - “1” — VCNEB方法;
 - “2” — 没有VCNEB计算的结构弛豫模型;
- Variable-Image-Number method:
 - “0” — VCNEB的Images数量固定;
 - “1” — VCNEB的Images数量可变;
- variability of spring constant:
 - “0” — 固定弹性常数;
 - “1” — 弹性常数可变;

Default: 110

Format:

111 : `vcnebType`

注意: 如果`vcnebType=111`, 即, VCNEB计算的Images值可变, 弹性常数可变。我们强烈建议用户用VCNEB方法研究重建相变时使用可变的Images值。

▷ variable `numImages`

Meaning: 执行计算初始的Images值。

Default: 9

Format:

13 : `numImages`

▷ variable `numSteps`

Meaning: 执行VCNEB计算的最大步数。

Default: 200

Format:

500 : numSteps

注意: (1) 当numSteps=-1, 初始路径仅在不执行能量计算时产生。(2) VCNEB路径通常收敛很慢。我们要求numSteps最少设为500。

▷ *variable* `optReadImages`

Meaning: Images文件的参考选项: :

- “0” — 所有images (numImages) 都需要尤其是Images文件;
- “1” — 只需要初始和最终的images且可以在Images中看的;
- “2” — 初始、最终和任何特定的中间Images都能在Images文件中看的。

Default: 2

Format:

1 : optReadImages

注意: 所有选项中, 初始和最终的images必须指定。自动线性插值法被应用于在选项1和2中产生初始Images。

▷ *variable* `optimizerType`

Meaning: 结构弛豫的优化算法选项:

- “1” — 急速下降 (SD);
- “2” — 快速惯性弛豫算法 (FIRE, Fast Inertial Relaxation Engine) 算法³²。

Default: 1 (SD) 对于VCNEB计算; 2 (FIRE) 对于结构弛豫

Format:

1 : optimizerType

▷ *variable* `optRelaxType`

Meaning: 结构弛豫类型:

- “1” — 仅弛豫原子位置 (晶胞固定), 跟传统NEB方法一样;
- “2” — 仅弛豫晶格 (仅用于测试);

- “3” — 原子位置和晶格完全弛豫。

Default: 3

Format:

3 : optRelaxType

▷ *variable* `dt`

Meaning: 结构弛豫的时间步数。

Default: 0.05

Format:

0.1 : dt

注意: 如果`dt`很小，计算就会很慢。如果`dt`很大，计算将不稳定且产生没有意义的路径。

▷ *variable* `ConvThreshold`

Meaning: images中均方根力（RMS, Root Mean Square forces）弛豫的标准条件。

Default: 0.003 eV/Å

Format:

0.005 : ConvThreshold

▷ *variable* `VarPathLength`

Meaning: 变Images方法的标准路径长度。当两个相邻images的长度比`VarPathLength`的1.5倍还大，新的image将用线性插入法加入两个images间；当小于0.5倍，第二个image将会移动。

Default: 初始Images间的平均路径长度

Format:

0.3 : VarPathLength

▷ *variable* `K_min`

Meaning: 最小弹性常数，仅用于变弹性常数VCNEB (单位 $\text{eV}/\text{\AA}^2$)。

Default: 5

Format:

3 : K_min

▷ *variable* `K_max`

Meaning: 最大弹性常数，仅用于变弹性常数VCNEB (单位 $\text{eV}/\text{\AA}^2$).

Default: 5

Format:

6 : K_max

▷ *variable* `Kconstant`

Meaning: 弹性常数，仅用于固定弹性常数VCNEB (单位 $\text{eV}/\text{\AA}^2$).

Default: 5

Format:

4 : Kconstant

▷ *variable* `optFreezing`

Meaning: 冻结Images结构的选项。当ConvThreshold被激活时Image结构将被冷冻:

- “0” — 任何时候都不冷冻Images。
- “1” — 当ConvThreshold被激活时冷冻Image。

Default: 0

Format:

1 : optFreezing

▷ *variable* `optMethodCIDI`

Meaning: Climbing-Image (CI) 和Downing-Image (DI) 方法的选项。这个方法只建议在你有一个合理且收敛的路径时使用:

- “0” — CI/DI 方法不用;
- “1” — 只用CI方法，只有最高能量或用户提供相变 (TS) 才用CI;
- “-1” — 只用DI方法，只有最低能量或用户提供相变 (LM) 才用DI;
- “2” — 混合多个-CI/DI方法，需提供FS和LM的连续数值。

Default: 0

Format:

1 : optMethodCIDI

▷ *variable* `startCIDISep`

Meaning: CI/DI法开始的步数，只有optMethodCIDI=1时可用。

Default: 100

Format:

```
200 : startCIDISep
```

▷ *variable* `pickupImages`

Meaning: Images ID为CI/DI-Image 法挑选。

Default: 过渡态Image ID和局域最小态Images

Format:

```
% pickupImages
9 11 17
% EndPickupImages
```

注意: 在这个例子中，第9、11、17个Images将被挑选应用CI/DI-Image法。过渡态的Image将应用CI-Image法，局域最小态Image将自动应用DI-Image法。

▷ *variable* `FormatType`

Meaning: 路径output文件中结构的格式，位于results1/PATH/. ?路径结构output格式:

- “1” — XCRYSDEN格式 (.xsf 文件) ;
- “2” —VASP POSCAR;
- “3” — 晶格的XYZ格式;

Default: 2

Format:

```
1 : FormatType
```

▷ *variable* `PrintStep`

Meaning: 保存在经过时间间隔后的每一步后位于results1/STEP/的VNCEB重新开始的文件。

Default: 1

Format:

```
10 : PrintStep
```

注意: 对于经验指令，如GULP，我们建议用户设PrintStep=10以减少保存新文件耗费的时间。

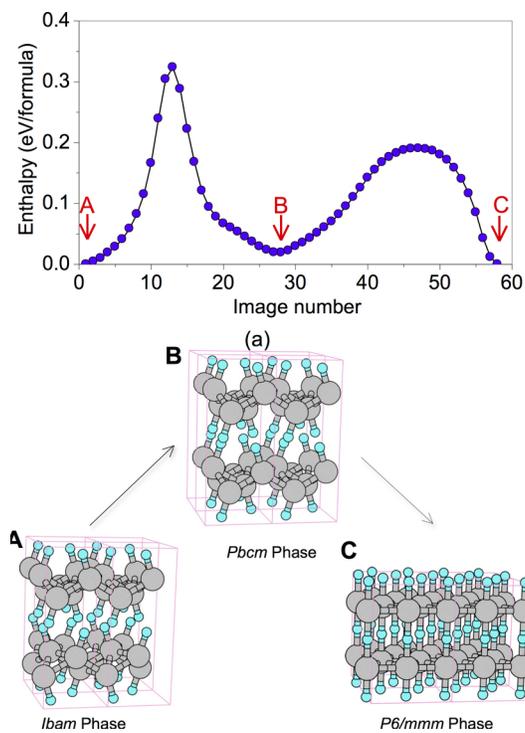


Figure 15: BH在168 GPa时发生相变¹⁴，出现Pbcm中间相。*Ibam*→*Pbcm*和*Pbcm*→*P6/mmm*片段上的鞍点的势垒分别为0.32、0.19 eV/f.u.

图Fig. 15展示了用VCNEB方法的例子：相变机制和能量势垒在BH体系在168 GPa发生*Ibam*→*P6/mmm*的转变，我们得到了Pbcm中间相。

6.3 如何设置VC-NEB计算的初始路径

VC-NEB方法是寻找相变路径非常有效的方法，但我们必须小心地准备初始路径。在VC-NEB计算中，晶胞旋转发生在初始和最后的结构附近，这里初始和最终的图像包含许多相同结构。为了排除这些无用的旋转，改进的Variable-Image-Number方法将自动出现晶胞旋转，这会节约很多时间。

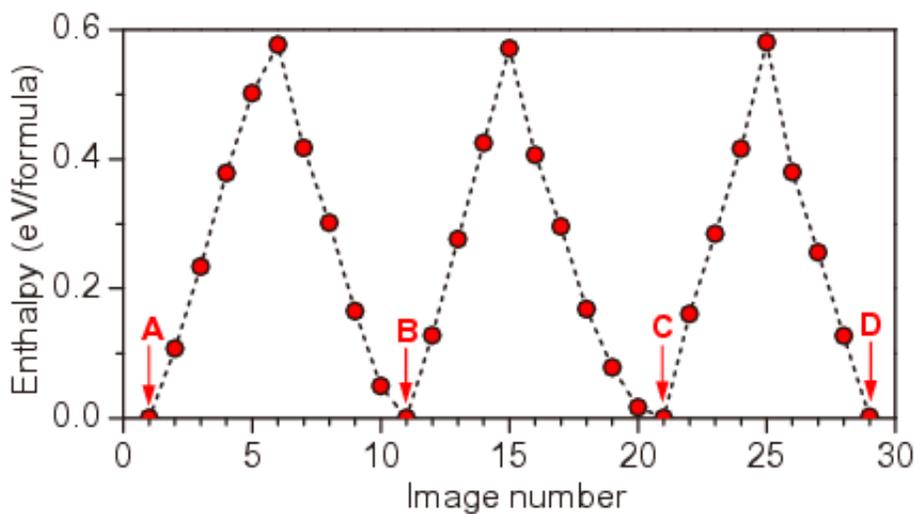
作为一种选择，当产生初始image设置时，可以在应用VC-NEB方法前采用拒绝循环技术。一般的 3×3 矩阵，欧拉角 $R(\phi, \theta, \psi)$ 和晶格反射 $M(x, y, z)$ 都被定义。执行VC-NEB计算前，全部空间的欧拉角和反射操作数值搜索都用来寻找最小晶胞转变距离 Δh ：

$$\Delta h = |h_{initial} - R(\phi, \theta, \psi)M(x, y, z)h_{final}|. \quad (9)$$

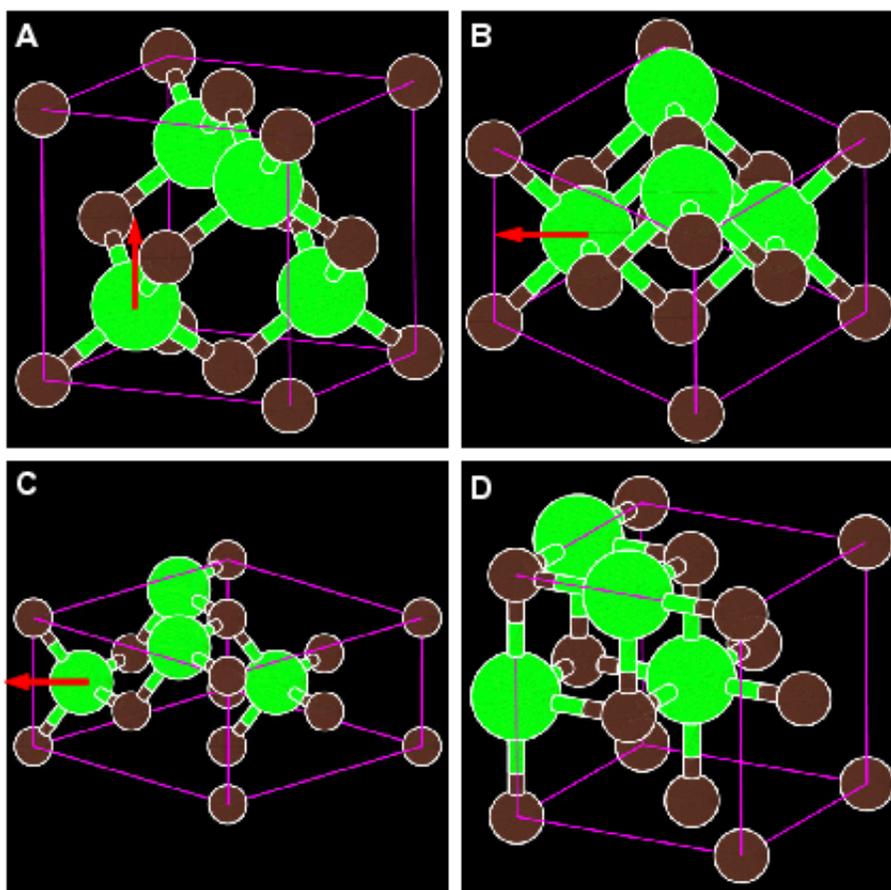
最后的image \tilde{h}_{final} 拒绝循环晶格适量被指定为终点的image：

$$\tilde{h}_{final} = R(\phi, \theta, \psi)M(x, y, z)h_{final}. \quad (10)$$

更重要的是，我们要预防任意分配初始和最终的images的原子部分坐标 \mathbf{r}_v （修改初始和最终结构的绘制原子）。否则，计算将会很困难或者一些相同的路径会出现在一个计算中，如图 16所示。对于更复杂的系统，如果你没有好的初始路径，将会得到一些不合理的或散乱的路径。在两个终点images上的原子最小距离的全局搜索帮助VC-NEB方法再分配原子序列。在VC-NEB计算前自动创造模型路径的能力对相转变的稳定和收敛非常重要，而且是研究大的复杂系统的先决条件。



(a)



(b)

Figure 16: 当设置“不好”的初始Image文件会发现相同的路径。路径是GaN在45 GPa静压下B3→B1的相变。在11和21Images, 单斜晶胞中的B1和B3结构在MEP搜索时被发现。Ga原子在相变过程中沿箭头方向移动。

7 在线工具

我们开发许多有用的在线工具，它们被用于准备USPEX的输入和后处理。查询这些工具请链接：

<http://han.ess.sunysb.edu>

下面你可以找到有关每个在线工具的信息。

7.1 结构表征

在这里我们有4个工具：

- **Fingerprints** — 计算和绘制指纹功能。这是一种晶体结构描述符。这一描述符是一种与对相关函数和衍射图案相关的一维函数。它不依赖于绝对的原子坐标，而只依赖于原子间的距离。原子位置的小偏差会轻微地影响指纹，即它们是数值稳健的。
- **Multifingerprint** — 计算平均准熵，以及关于一系列结构的一阶和S阶。它通过余弦距离差 ≥ 0.003 这一条件过滤出独特的结构，识别出这些结构的对称性，最后在`uniq_gatheredPOSCARS`文件中将它们一一列举出来。
- **POSCAR2CIF** — 确定了空间群和从POSCAR文件中准备一个CIF文件。
- **XSF2POSCAR** — 从XSF (XCRYSDEN) 文件中准备一个POSCAR文件。

7.2 性质计算

这里我们有2个工具：

- **Hardness** — 这一工具基于Lyakhov-Oganov模型计算硬度。
- **EELS** — 这一工具用于计算电子能量损失谱，EELS是Priya Johari编写的。

7.3 分子晶体

在这里我们有2个工具：

- **MOL precheck** — 该工具允许你在运行含有`calculationType=310/311/110`的USPEX计算器之前查阅MOL_1文件。
- **Zmatrix** — 该工具可将XYZ文件转换为USPEX MOL_1文件。

7.4 表面

[Substrate](#) — 基底——一个从POSCAR/CIF文件和指定的密勒指数，层厚度和转变中准备基底的程序。由此产生的POSCAR文件可用于`calculationType=200/201`作为基底的表面计算。

7.5 其它

我们有以下工具:

- [Input generator](#) — USPEXINPUT.txt生成器。该实用程序可以帮助初学者建立正确的USPEX输入文件。
- [Volume estimation](#) — 对于USPEX（或对于INPUT.txt文件）该功能可以估计非分子晶体与分子晶体的体积。
- [USPEX manual](#) — 本手册的在线版本。
- [USPEX examples](#) — archives with USPEX examples for versions 9.3.9 and 9.4.2.

8 常见问题

8.1 如何将结果可视化？

USPEX会产生大量数据（晶体结构和能量等等）。通过“手动”分析这些数据将会非常繁琐和耗时。通过采用高效可视化，分析可以更迅速地进行，可以产生有价值的新见解。Mario Valle使用他的STM4可视化工具包³³已经制定了具体的读取功能和可视化USPEX输出文件，在这里我们强烈建议使用与USPEX相匹配的工具包。使用STM4，需要在计算机上安装AVS / Express。AVS / Express不是公开发行的，所以你需要一个许可证。或者，你可以使用其他软件，例如OpenDX或VESTA进行USPEX结果的可视化。VESTA已经包含了直接读取USPEX结果文件的功能。图Fig. 17显示了由STM4产生的典型图谱，此外STM4还可以制作影像，用以分析电荷密度图，你可以在<http://mariovalle.name/STM4>里找寻应用。

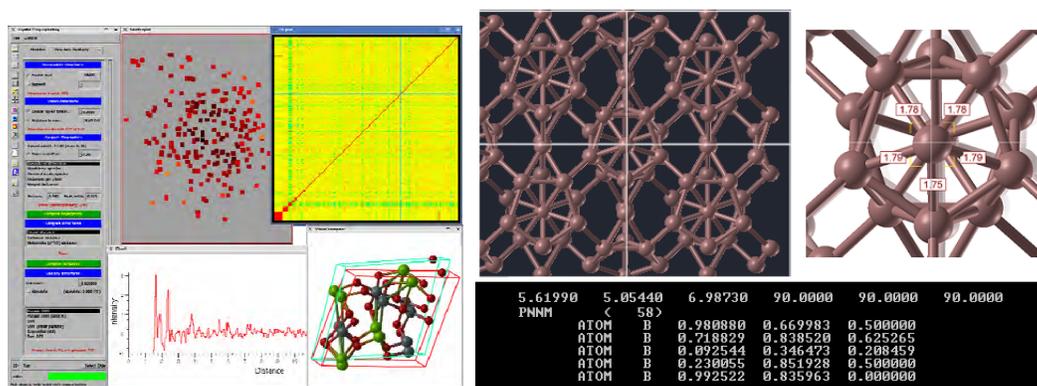


Figure 17: STM4的USPEX界面。

另外，对于USPEX结果的可视化，也可以使用OpenDX, VESTA,等其他软件。VESTA也已经添加了直接读取USPEX结构文件的功能。

8.2 如何避免陷阱？

第一，USPEX使用一个足够大的种群规模。第二，USPEX通过默认采用一种强大的指纹识别方法。任何增加种群多样性的因素会减少受困于局部最小值的几率。为了确保模拟不陷入困境，运行一个带有不同参数的二次模拟是有必要的。反种子技术是避免陷入困境的一个强大的方式。

8.3 什么是单块计算？

在USPEX 9.3.9中对单块特征进行了介绍，单块特征使USPEX 用户能够利用相同混合物可变数目的计算单位进行结构预测。例如：

```
% atomType
Si 0
% EndAtomType

% numSpecies
1 2
% EndNumSpecies

12 : minAt
24 : maxAt
```

这意味着我们取样的带有可变数目计算单位的SiO₂化合物（按1:2的比例）具有12–24个原子。

从USPEX 9.4.1开始，单块功能已经从`calculationType = 301/311`变为`300/310`。仍然是相同的设置，用户只需要设置`minAt`、`maxAt`和`numSpecies`关键词。

8.4 如何使用种子技术？

这种技术是有用的，如果不是从随机结构开始，你会更愿意输入一些你已经知道的化合物或相关材料结构。在VASP5格式中的级联POSCAR文件的格式如下，为下一代结构创建Seeds/POSCARS文件，或为一次USPEX计算的特殊代创建Seeds/POSCARS_gen(gen指代号)。在此文件名中不要遗忘字母“S”。

例如：

```
EA33 2.69006 5.50602 4.82874 55.2408 73.8275 60.7535 no SG
1.0
2.690100 0.000000 0.000000
2.690100 4.804100 0.000000
1.344900 2.402100 3.967100
Mg Al O
1 2 4
Direct
0.799190 0.567840 0.859590
0.793520 0.230950 0.544750
0.793540 0.916090 0.174450
0.050972 0.816060 0.859610
0.172230 0.194810 0.859600
0.438250 0.655170 0.406880
0.438230 0.202440 0.312330
EA34 7.61073 2.85726 2.85725 60.0001 79.1809 79.1805 no SG
1.0
7.610700 0.000000 0.000000
0.536350 2.806500 0.000000
0.536330 1.352000 2.459300
Mg Al O
1 2 4
Direct
```

```

0.708910  0.507440  0.068339
0.374050  0.285730  0.846630
0.023663  0.069185  0.630090
0.889560  0.780560  0.341460
0.350470  0.626920  0.187820
0.597290  0.211310  0.772210
0.116440  0.371590  0.932500

```

在计算中的任何时候都可以加入种子。每一代开始时USPEX将尝试读取种子文件的两个类型。相应的信息将被记录在results1/Seeds_history中，种子文件（POSCARS或POSCARS_gen）将为被作为POSCARS_gen保存在种子Seeds/文件中。

当种子加入时，我们建议用户检查results1/Seeds_history和警告文件。如果你的种子有问题时，可能会有一个警告信息“Meet a problem when reading Seeds - ...”。当一个错误出现在种子文件，比如是缺失了一行，错误出现后的结构将不会被添加进去。

注意：确保你指定的所有原子符号在每一结构的6号线位置。例如，添加P6₃/mc H₂ 结构到H-O变成分计算，你要编辑的文件：

```

H_I-P63/mc
1
4.754726  -2.74514  0.000000
-0.000000  5.490285  0.000000
0.000000  0.000000  4.508715
H
16
Direct
...
(the atomic positions information is omitted here)

```

8.5 如何巧妙考虑组成？

对于可变组分和单块计算，当计算开始时，它便会生成一个带有所有可能组成的种子文件Seeds/compositions，其中遗传代码从随机结构发生器中随机选取组成。例如，C-O体系：

```

8 0
0 8
2 4

```

意味着你会对随机产生C₈、O₈和C₂O₄结构怀有兴趣。由于遗传变异和原子置换，其他的成分也将被采样。

当想要产生含有指定组成的结构时，可以使用anticompositions 的特点—将所有不想要的组成输入至Seeds/Anti-compositions文件中。主要有3种可行方式：

1. 对于所有同一比例不需要的成分，可以写下某一成分比以禁止某些成分。例如，可以使用“1 2 0”禁止所有的以相同的比例组成的成分，如“1 2 1”、“2 4 2”和“3 6 3”等。
2. 只为特定的成分，而不是其他具有同一成分比的组成。可以用一个负号标识组成。例如，可以使用“-3 2 0”或“3 -2 0”禁止“3 2 0”的成分，但不禁止“6 4 0”或“9 6 0”组成。（注意：“3 2 -0”这种情况下无意义）。
3. 对于所有的单一/二元/三元化合物，如果你不想对所有的单一/二元/三元化合物进行采样，请在Anti-compositions文件中写下关键词single/binary/ternary。

例如：

```
single
binary
1 1 2
-2 2 1
```

如果你不太确定你正在做什么，就让Anti-compositions文件空着吧。为了得到你不想要的组分的信息，你可以查看下results1/compositionStatistic。

注意：

- 即使在计算开始之前compositions或Anti-compositions文件已存在，他们会被忽略。Anti-compositions文件会被重命名为Anti-compositions-back备份文件。因此，在计算开始之后请编辑compositions或Anti-compositions文件。
- 也请注意，在带有组合块的USPEX计算中，组合物通常意味着这些块的数量。因此，为了确保Anti-compositions文件的格式正确，请首先检查组成compositions文件。

8.6 如何建立一个从本地机器到远程集群的密码连接？

有两种方式来解决这个问题：

1. 不需要密码的SSH登录。（注意，本小结基于文章http://linuxproblem.org/art_9.html）
你的目标：你想使用OpenSSH进行作业地自动提交，因此你需要一个从hostA或者userA到hostB或者userB的自动登录。在这个过程中你不想输入任何密码，因为你需要一个shell脚本去激活ssh。

如何去做：首先以userA的身份登录hostA，接着产生一对验证密钥。不需要输入密码：

```
userA@hostA:~> ssh-keygen -t rsa Generating public/private rsa key
pair.
Enter file in which to save the key (/home/userA/.ssh/id_rsa):
Created directory '/home/userA/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/userA/.ssh/id_rsa.
Your public key has been saved in /home/userA/.ssh/id_rsa.pub.
The key fingerprint is:
3e:4f:05:79:3a:9f:96:7c:3b:ad:e9:58:37:bc:37:e4 userA@hostA
```

然后以userB的身份登录hostB，使用ssh去产生一个目录（目录可能本身已经存在，但是没关系）

```
userA@hostA:~> ssh -p portB userB@hostB 'mkdir -p .ssh' userB@hostB's
password:
```

最后将A的新公共密钥附加给hostB的userB：.ssh/authorized_keys 同时最后一次输入B的密码：

```
userA@hostA:~> cat .ssh/id_rsa.pub | ssh -p portB userB@hostB'cat >>
.ssh/authorized_keys'.
userB@hostB's password:
```

从现在开始，你就可以避免从hostA里userA的身份登录hostB里userB时输入密码。

```
userA@hostA:~> ssh -p portB userB@hostB
```

2. 你需要复制从本地机器到远程集群的公共密钥（目录./ssh或./ssh2）。这里是你需要执行的命令列表：

```
local # ssh-keygen -t dsa
local # scp ~/.ssh2/id_dsa.pub oganov@palu.cscs.ch:~/.ssh/tmp.pub
remote # cd ~/.ssh/
remote # ssh-keygen -f tmp.pub -i >> authorized_keys
remote # rm tmp.pub
```

8.7 如何从一个损坏了的*.mat文件重新开始计算？

当在文件系统中存在一个问题，例如磁盘已满，文件系统超载，USPEX难以准确写下这些*.mat文件，在计算过程中便会出现如下信息：

```
???Error using ==> load
Unable to read MAT file /home/USPEX/Current_POP.mat

File may be corrupt.
```

当遇到一个坏了的*.mat文件引起的错误时，请利用备份恢复损毁的*.mat文件，删除matfilelocker文件，然后重新启动计算。不幸地，若是当备份*.mat文件是空的或还是错误时，你必须重新在这一代中重新启动计算。

8.8 当USPEX在一段时间内不运行，该如何操作？

当发现USPEX在一段时间里不运行，而文件still_running仍存在了较长的时间（通常超过30分钟，请使用命令“ls -l”以检查仍在运行文件的时间戳），你应该考虑到该计算存在某些错误。在这种情况下，你需要做的是遵循以下的检查程序：

- 使用命令“top”检查确保您的MATLAB计算没有运行，有时，USPEX在结构的生成和软模变异上会花费较长时间。一旦你确定MATLAB停止，你可以开始下一步。
- 停止crontab或作业运行脚本以避免在检查程序过程中USPEX运行。这是非常重要的，否则你会搞砸了你的USPEX计算。
- 删除still_running文件。
- 输入命令“USPEX -r”或“matlab < USPEX.m”运行USPEX看会发生什么。如果你遇到错误或缺陷，你可以尝试修复它们，或是向我们的谷歌USPEX论坛寻求帮助，或让我们知道，并报告缺陷。
- 如果一切都很好，只需要重新启动crontab或作业运行脚本继续计算。

8.9 如何建立一个作业提交脚本？

为了建立一个工作提交脚本，我们希望用户了解关于MATLAB编程和你的作业提交系统的一些基本知识，至少要明白如何处理MATLAB中的字符串和如何获得工作信息的基本思路。

有两种工作脚本提交模式：**本地提交**或**远程提交**，这取决于你是将*ab initio*计算提交到你运行USPEX和MATLAB的本地机器还是远程超级计算机。

8.9.1 第一步：在提交文件夹Submission/中配置文件

案例一：本地提交。

请在INPUT.txt文件中编辑下列标签：

```
1 : whichCluster (0: no-job-script, 1: local submission, 2: remote submission)
```

然后到Submission/文件夹下，这里你需要编辑submitJob_local.m和checkStatus_local.m这两个文件。

你可以在这些文件中找到详细的说明。在一般情况下，你只需告诉USPEX如何提交工作和检查工作是否已经完成。

在submitJob_local.m里：

```

1 function jobNumber = submitJob_local()
2 %
3 %This routine is to check if the submitted job is complete or not
4 %One needs to do a little edit based on your own situation.
5 %1 : whichCluster (default 0, 1: local submission, 2: remote submission)
6 %
7
8 %Step 1: to prepare the job script that is required by your supercomputer
9 fp = fopen('myrun', 'w');
10 fprintf(fp, '#!/bin/sh\n');
11 fprintf(fp, '#PBS -l nodes=1:ppn=8,walltime=1:30:00 -q cfn_short\n');
12 fprintf(fp, '#PBS -N USPEX\n');
13 fprintf(fp, '#PBS -j oe\n');
14 fprintf(fp, '#PBS -V \n');
15 fprintf(fp, 'cd ${PBS_O_WORKDIR}\n');
16 fprintf(fp, 'mpirun -np 4 vasp1 > vasp1.out\n');
17 fclose(fp);
18
19 %Step 2: to submit the job with a command like qsub, bsub, llsubmit, etc.
20
21 [a,b]=unix(['qsub myrun'])
22
23 %Step 3: to get the jobID from the screen message
24 %It will output some message on the screen like '2350873.nano.cfn.bnl.local'
25
26 end$_marker$ = findstr(b, '.');
27 jobNumber = b(1:end$_marker$(1)-1);

```

案例二：远程提交。

请在INPUT.txt文件中编辑下列标签：

```

2 : whichCluster (default 0, 1: local submission; 2: remote submission)
C-20GPa : remoteFolder

```

最后到Submission/文件夹下，此处你需要编辑两个文件：

submitJob_remote.m和checkStatus_remote.m。

In submitJob_remote.m:

```

1 function jobNumber = submitJob_remote(USPEX, Index)
2 %-----
3 %This routine is to check if the submitted job is complete or not
4 %2 : whichCluster (default 0, 1: local submission; 2: remote submission)
5 %C-20GPa : remoteFolder
6 %-----
7
8 %-----
9 %Step1: To prepare the job script, runvasp.sh
10 fp = fopen('runvasp.sh', 'w');
11 fprintf(fp, '#!/bin/sh\n');
12 fprintf(fp, '#PBS -l nodes=2:ppn=2,walltime=1:30:00\n');
13 fprintf(fp, '#PBS -N USPEX\n');
14 fprintf(fp, '#PBS -j oe\n');
15 fprintf(fp, '#PBS -V \n');
16 fprintf(fp, 'cd ${PBS.O_WORKDIR}\n');
17 fprintf(fp, '/usr/local/pkg/openmpi-1.4.5/bin/mpirun -np 4 vasp1 > vasp.out\n');
18 fclose(fp);
19 %-----
20 %Step 2: Copy the files to the remote machine
21
22 %Step2-1: Specify the PATH to put your calculation folder
23 Home = ['/nfs/user08/qiazhu']; %'pwd' of your home directory on remote machine
24 Address = 'qiazhu@seawulf.stonybrook.edu'; %your target server: username@address
25 Path = [Home '/' USPEX '/CalcFold' num2str(Index)]; %Just keep it
26
27 %Step2-2: Create the remote directory
28 % Please change the ssh/scp command if necessary!
29 % Sometimes you don't need the -i option
30 try
31 [a,b]=unix(['ssh -i ~/.ssh/seawulf ' Address ' mkdir ' USPEX ]);
32 catch
33 end
34
35 try
36 [a,b]=unix(['ssh -i ~/.ssh/seawulf ' Address ' mkdir ' Path ]);
37 catch
38 end
39
40 %Step2-3: Copy the necessary files (for VASP calculations, we need POSCAR, INCAR, POTCAR,
41 % KPOINTS and job script)
42 unix(['scp -i ~/.ssh/seawulf POSCAR ' Address ':' Path]);
43 unix(['scp -i ~/.ssh/seawulf INCAR ' Address ':' Path]);
44 unix(['scp -i ~/.ssh/seawulf POTCAR ' Address ':' Path]);
45 unix(['scp -i ~/.ssh/seawulf KPOINTS ' Address ':' Path]);
46 unix(['scp -i ~/.ssh/seawulf runvasp.sh ' Address ':' Path]);
47
48 %-----
49 %Step 3: to submit the job and get JobID, i.e., the exact command to submit the job.
50 [a,v]=unix(['ssh -i ~/.ssh/seawulf ' Address ' /usr/local/pkg/torque/bin/qsub '
51 Path '/runvasp.sh'])
52
53 % format: Job 1587349.nagling is submitted to default queue <mono>
54 end_marker = findstr(v, '.');
55 if strfind(v, 'error')
56 jobNumber=0;
57 else
58 jobNumber = v(1:end_marker(1)-1);
59 end

```

在CheckStatus_remote.m里:

```

1 function doneOr = checkStatus_remote(jobID, USPEX, Folder)
2 %-----
3 %This routine is to check if the submitted job is complete or not
4 %One needs to do a little edit based on your own situation.
5 %-----
6
7 %Step1: Specify the PATH to put your calculation folder
8 Home = ['/nfs/user08/qiazhu']; %'pwd' of your home directory of your remote machine
9 Address = 'qiazhu@seawulf.stonybrook.edu'; %Your target: username@address.
10 Path = [Home '/' USPEX '/CalcFold' num2str(Folder)]; %just keep it
11 %Step2: Check JobID, the exact command to check job by jobID
12 [a,b]=unix(['ssh -i ~/.ssh/seawulf ' Address ' /path/to/qstat ' num2str(jobID)])
13     tempOr1 = strfind(b, 'R batch');
14     tempOr2 = strfind(b, 'Q batch');
15     if isempty(tempOr1) & isempty(tempOr2)
16         doneOr = 1;
17 % for vasp, we usually need OSZICAR for reading energy and CONTCAR for reading
18 %structure OUTCAR, EIGENVAL, DOSCAR might be needed for reading other properties.
19 % unix(['scp -i ~/.ssh/seawulf ' Address ':' Path '/OUTCAR ./'])
20 %OUTCAR is not necessary by default
21     unix(['scp -i ~/.ssh/seawulf ' Address ':' Path '/OSZICAR ./'])
22 %For reading enthalpy/energy
23     unix(['scp -i ~/.ssh/seawulf ' Address ':' Path '/CONTCAR ./'])
24 %For reading structural info
25 end

```

这可能需要一些时间来正确配置这些文件。为了测试它是否可以工作，你可以输入两次“USPEX -r”指令，然后监测屏幕信息。第一次的尝试是检查工作是否提交，而第二次是检查USPEX是否可以正确检测提交工作的状态。所有的相关信息，可以在屏幕上输出信息中找到。如果MATLAB输出没有任何错误，你可以准备好离开了。

8.9.2 第二步：定期运行USPEX

实际计算以“USPEX -r > log”命令开始。每次MATLAB程序都会检查*ab initio* 计算的运行状态。如果工作完成，MATLAB就会前往计算文件夹读取结果，随后提交新的计算。之后，MATLAB将退出。因此，需要定期调用命令（例如，每5分钟）。周期性的脚本可以通过使用crontab或shell脚本被执行。

8.9.3 Crontab

这可以在你的Linux 机器上执行一个crontab后台程序。在你的用户主目录中，现在应该有这些文件：

```
~/call_job
~/CronTab
```

下面是来自我们的集群的1-line CronTab文件例子的一行：

```
1 */5 * * * * sh call_job
```

它表示作业提交之间的间隔为5分钟，并指向文件`call_job`，它应该包含USPEX将被执行目录的地址，`call_job`大概看起来是这样：

```
1 #!/bin/sh
2 source $HOME/.bashrc
3 cd /ExecutionDirectory
4 date >> log
5 USPEX -r >> log
```

为了激活`crontab`，键入

```
1 crontab ~/CronTab
```

如果你想终止运行，或编辑`call_job`文件或通过键入以下命令删除`crontab`

```
1 crontab -r
```

检查`crontab` 是否工作正常，你应该在计算开始时追踪日志文件的更新。

8.9.4 Shell脚本

可以在Linux shell中使用`sleep`睡眠命令准备脚本。下面是一个非常简单的脚本`run-uspex.sh`：

```
1 #!/bin/sh
2 while [ ! -f ./USPEX_IS_DONE ]; do
3     date >> log
4     USPEX -r >> log
5     sleep 300
6 done
```

注意： 记住这个计算可以通过终止该脚本的进程ID来停止。

8.10 如何使对称性代码在32位机器上工作？

在`FunctionFolder/Tool/32bit/`下执行“`./install-32bit.sh`” 来使32位可执行文件替换默认的64位二进制可执行。

9 附录

9.1 例子目录

- EX01-3D_Si_vasp: 零压力下的硅（每个单胞中含8个原子）。使用VASP, PBE96泛函进行变胞DFT计算。感谢G. Kresse和他的PAW文件（POTCAR）对我们的贡献。
- EX02-3D_MgAl2O4_gulp: 在100 GPa 下的MgAl₂O₄（每一个单胞中有28个原子）。使用Buckingham势，GULP代码进行变胞计算。如果需要很可靠的结果，最好用*ab initio*算法。
- EX03-3D-const_cell_MgSiO3_gulp: 这个例子向我们展示在知道单胞参数时如何进行结构预测。含有Buckingham势，GULP 代码的MgSiO₃（每个单胞有20个原子）。晶胞参数与后钙钛矿一致。后钙钛矿的发现（*Oganov & Ono, Nature 2004; Murakami et al., Science 2004*）在地球科学领域是一个重大突破。
- EX04-3D_C_lammps: 这个例子展示了如何使用结合了LAMMPS码的USPEX进行晶体结构预测。在这个简单的例子中：有8个碳原子，采用Tersoff势。
- EX05-3D_Si_atk: 这是一个使用密度泛函紧束缚近似和ATK 代码对每一单胞含8个原子的硅进行晶体结构预测的例子。
- EX06-3D_C_castep: 在0 GPa下，使用CASTEP代码对每个单胞含8个原子的碳的晶体结构进行基于DFT的预测。
- EX07-2D_Si_vasp: 使用DFT和VASP预测硅的二维晶体。简单而有效。
- EX08-0D_LJ_gulp: 纳米结构预测。含有30个原子的Lenard-Jones纳米颗粒，GULP代码。
- EX09-3D-molecules_CH4_vasp: 甲烷分子晶体预测，压力20 GPa，DFT，VASP。分子在文件MOL_1中描述。
- EX10-3D-molecules_CH4_dmacrys: 常压，带有8个分子的单胞，采用力场和DMACRYS代码。在文件MOL_1中描述分子，但要注意其DMACRYS计算略有不同的格式。请把可执行文件dmacrys、neighcrys-pp和neighcrys-vv放在文件夹Specific/。
- EX11-3D-molecules_urea_tinker: 常压，2个分子的单胞，采用力场和TINKER代码。在文件MOL_1中描述分子。
- EX13-3D_special_quasirandom_structure_TiCoO: 含有伪“Mo”和“B”原子，GULP和变组成USPEX（*Lyakhov and Oganov, 2010*）的Lenard-Jones二元体系。
- EX13-3D_special_quasirandom_structure_TiCoO: USPEX可以很容易地找到最无序（或最有序）合金结构。这里表示为Ti_xCo_(1-x)O。你需要在Seeds/POSCARS

指定初始结构和仅使用置换操作。在这种情况下，不需要使用任何外部代码。在这个例子中，我们在无弛豫的情况下(`abinitioCode = 0`)优化（最小化）结构有序度（*Oganov and Valle (2009); Lyakhov, Oganov, Valle (2010)*）。种子结构（Ti-Co-O结构的超晶胞）被置换找到结构的最小/最大化有序度。在这种情况下最小化有序度，可以得到“特殊准随机结构”的通用版本。

- **EX14-GeneralizedMetadynamics_Si_vasp**: 关于开始用一个简单的种子结构搜索复杂低能量结构的简单例子（*Zhu et al, 2013*）。硅，最多为16个原子的单胞，DFT，VASP。要特别注意INCAR文件。最重要的是，只要保留你在这里看到的文件，只改变ENCUT，SIGMA，进化准动力学不仅预测低能量的结构，同时也给出了晶体结构之间的转换机制。
- **EX15-VCNEB_Ar_gulp**: 氩气，在0 GPa压力模型系统中采用变胞微动弹性带（*VCNEB: Qian et al., 2013*）计算fcc-hcp转变。Lenard-Jones势，GULP代码。
- **EX16-USPEX-performance_SrTiO3_gulp**: SrTiO₃（50原子/单胞）在零压力下。使用Buckingham势，GULP代码进行可变单胞计算。从这个例子中可以看到，即使是这样一个比较大的系统，USPEX代码仍有>90%的成功率和显著的效率。与此相反，通过Zurek和Lonie使用相同的体系，相同的势能报道的成功率只有7-12%。显然，USPEX优于我们通过Zurek和Lonie的实现方法。我们已经见证了我们的代码的性能在更大的系统也很优异。
- **EX17-3D_DebyeTemp_C_vasp**: 对弹性相关性能优化的例子（体积和剪切模量，泊松比，Chen-Niu硬度，德拜温度）在这个例子中，我们使用VASP代码最大限度地提高碳的德拜温度。
- **EX18-3D_varcomp_ZnOH_gulp**: 正如你所知道的，USPEX对于变成分的计算具有独特的能力。这个例子为我们展现了一个非常具有挑战性的情况—对于三元体系Zn-O-H的变成分计算。这个计算在GULP代码中使用ReaxFF力场。USPEX可以为任何成分组成进行计算—例如，在其它范围内可实现的四元和五元体系。当然，成分越多，计算越昂贵（和越有风险）。目前没有任何参考结果。
- **EX19-Surface-boron111**: 预测alpha-B的(111)表面重构，采用变原子数(*Zhou et al., Phys. Rev. Lett. 113, 176101 (2014)*)。
- **EX20-MinHop_SiO2_gulp**: SiO₂(300)的最小跳跃法计算。
- **EX21-SingleBlock_Magnetic_Fe3C_VASP**: 在单块模式(300)中使用VASP预测Fe₃C磁性结构的预测。
- **EX22-3D_Cluster_C60_MOPAC**: 使用MOPAC进行C₆₀团簇结构的预测(000)。

9.2 测试运行

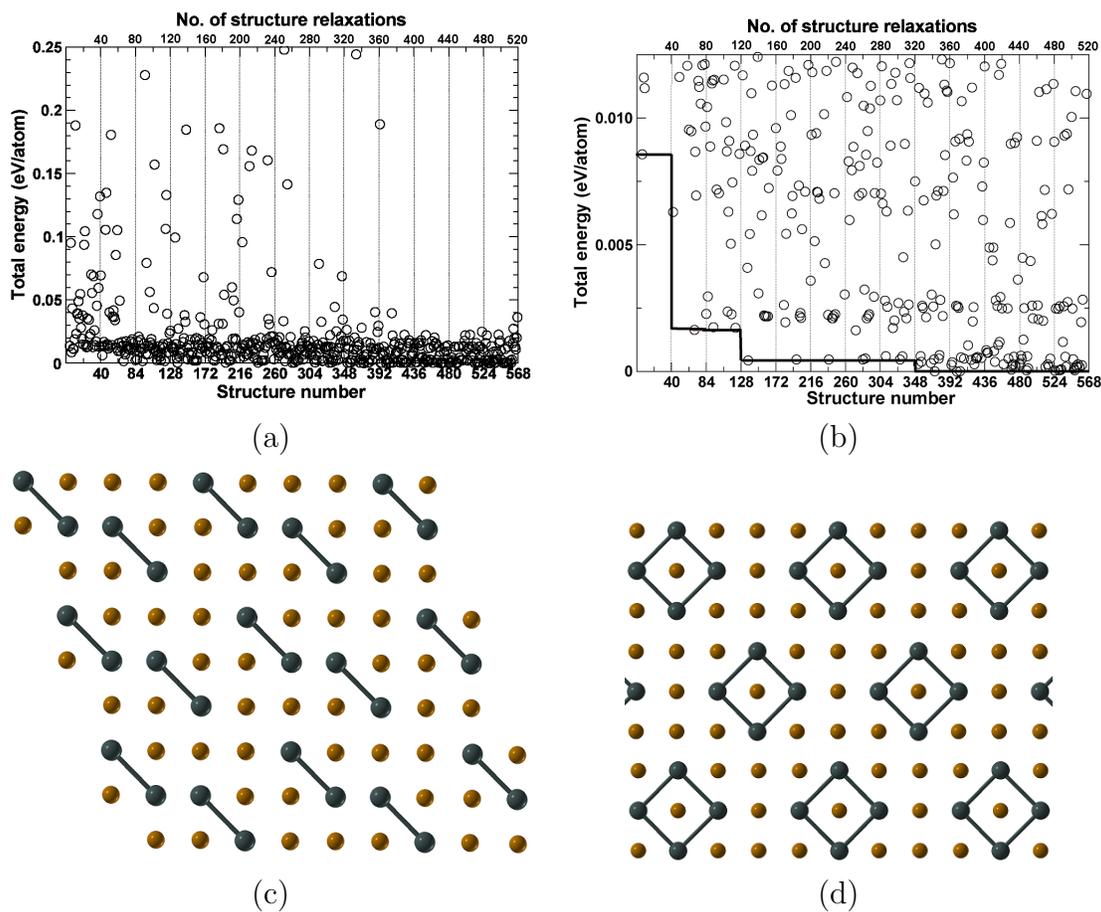


Figure 18: 对 Au_8Pd_4 进化结构搜索。a, b—总能量进化（为了清晰，（b）可以在同一数据集能量最低的区域急剧上升），c—在我们的进化模拟中发现的能量最低结构，d—利用准噶尔集群扩展的计算发现的能量最低结构。请注意我们的结构（c）是这种化合物已知的最低能量结构。这建立了我们方法的权威性（即使是旧的2007版）。

9.3 INPUT.txt输入文件示例

9.3.1 Fixed-composition USPEX calculation (calculationType=300):

```
PARAMETERS EVOLUTIONARY ALGORITHM
2 % Example of the short input, using most options as defaults
4 % atomType
Mg Al O
6 % EndAtomType
8 % numSpecies
2 4 8
10 % EndNumSpecies
12 50 : numGenerations
50.0 : ExternalPressure
14
16 % abinitioCode
3 3 3 3 3
% ENDabinit
18
20 % commandExecutable
gulp < input > output
% EndExecutable
```

9.3.2 Variable-composition USPEX calculation (calculationType=301):

```
1 USPEX : calculationMethod (USPEX, VCNEB, META)
2 301   : calculationType (dimension: 0-3; molecule: 0/1; varcomp: 0/1)
3 1     : AutoFrac
4
5 % atomType
6 Mo B
7 % EndAtomType
8
9 % numSpecies
10 1 0
11 0 1
12 % EndNumSpecies
13
14 80    : populationSize
15 200   : initialPopSize
16 60    : numGenerations
17 20    : stopCrit
18
19 11    : firstGeneMax
20 8     : minAt
21 18    : maxAt
22
23 % abinitioCode
24 3 3 3
25 % ENDabinit
26
27 % commandExecutable
28 gulp < input > output
29 % EndExecutable
```

9.3.3 Evolutionary metadynamics (calculationMethod=META):

```
1 META : calculationMethod (USPEX, VCNEB, META)
2 301 : calculationType (dimension: 0-3; molecule: 0/1; varcomp: 0/1)
3
4 % valences
5 4
6 % endValences
7
8 % IonDistances
9 1.2
10 % EndDistances
11
12 0.0001 : ExternalPressure
13
14 16 : maxAt
15 2.0 : minVectorLength
16 8.0 : maxVectorLength
17
18 15 : populationSize
19 40 : numGenerations
20 3.0 : mutationDegree
21 250.0 : GaussianHeight
22 0.3 : GaussianWidth
23 2 : FullRelax
24
25 abinitioCode
26 1 1 1 (1 1)
27 ENDabinit
28
29 % KresolStart
30 0.12 0.10 0.09 0.10 0.08
31 % Kresolend
32
33 % commandExecutable
34 mpirun -np 4 vasp > log
35 % EndExecutable
```

9.3.4 vcNEB calculation (calculationMethod=VCNEB):

```
1 VCNEB : calculationMethod
3 % numSpecies
4
5 % EndNumSpecies
7 % atomType
8 Ar
9 % EndAtomType
11 0.0 : ExternalPressure
13 111 : vcnebType
14 15 : numImages
15 500 : numSteps
16 1 : optimizerType
17 2 : optReadImages
18 3 : optRelaxType
19 0.25 : dt
20 0.003 : ConvThreshold
21
22 0.3 : VarPathLength
23 3 : K_min
24 6 : K_max
25 0 : optFreezing
26 0 : optMethodCIDI
27
28 2 : FormatType
29 10 : PrintStep
31 abinitioCode
32 3
33 ENDabinit
35 % commandExecutable
36 gulp < input > output
37 % EndExecutable
```

9.4 空间群列表

1	<i>P1</i>	2	<i>P-1</i>	3	<i>P2</i>	4	<i>P21</i>
5	<i>C2 (A2)*</i>	6	<i>Pm</i>	7	<i>Pc (Pa)*</i>	8	<i>Cm (Am)*</i>
9	<i>Cc (Aa)*</i>	10	<i>P2/m</i>	11	<i>P21/m</i>	12	<i>C2/m (A2/m)*</i>
13	<i>P2/c (P2/a)*</i>	14	<i>P21/c (P21/a)*</i>	15	<i>C2/c (A2/a)*</i>	16	<i>P222</i>
17	<i>P2221</i>	18	<i>P21212</i>	19	<i>P212121</i>	20	<i>C2221</i>
21	<i>C222</i>	22	<i>F222</i>	23	<i>I222</i>	24	<i>I212121</i>
25	<i>Pmm2</i>	26	<i>Pmc21</i>	27	<i>Pcc2</i>	28	<i>Pma2</i>
29	<i>Pca21</i>	30	<i>Pnc2</i>	31	<i>Pmn21</i>	32	<i>Pba2</i>
33	<i>Pna21</i>	34	<i>Pnn2</i>	35	<i>Cmm2</i>	36	<i>Cmc21</i>
37	<i>Ccc2</i>	38	<i>Amm2 (C2mm)*</i>	39	<i>Aem2 (C2mb)*</i>	40	<i>Ama2 (C2cm)*</i>
41	<i>Aea2 (C2cb)*</i>	42	<i>Fmm2</i>	43	<i>Fdd2</i>	44	<i>Imm2</i>
45	<i>Iba2</i>	46	<i>Ima2</i>	47	<i>Pmmm</i>	48	<i>Pnnn</i>
49	<i>Pccm</i>	50	<i>Pban</i>	51	<i>Pmma</i>	52	<i>Pnna</i>
53	<i>Pmna</i>	54	<i>Pcca</i>	55	<i>Pbam</i>	56	<i>Pccn</i>
57	<i>Pbcm</i>	58	<i>Pnmm</i>	59	<i>Pmmn</i>	60	<i>Pbcn</i>
61	<i>Pbca</i>	62	<i>Pnma</i>	63	<i>Cmcm</i>	64	<i>Cmce (Cmca)*</i>
65	<i>Cmmm</i>	66	<i>Cccm</i>	67	<i>Cmme (Cmma)*</i>	68	<i>Ccce (Ccca)*</i>
69	<i>Fmmm</i>	70	<i>Fddd</i>	71	<i>Immm</i>	72	<i>Ibam</i>
73	<i>Ibca</i>	74	<i>Imma</i>	75	<i>P4</i>	76	<i>P41</i>
77	<i>P42</i>	78	<i>P43</i>	79	<i>I4</i>	80	<i>I41</i>
81	<i>P-4</i>	82	<i>I-4</i>	83	<i>P4/m</i>	84	<i>P42/m</i>
85	<i>P4/n</i>	86	<i>P42/n</i>	87	<i>I4/m</i>	88	<i>I41/a</i>
89	<i>P422</i>	90	<i>P4212</i>	91	<i>P4122</i>	92	<i>P41212</i>
93	<i>P4222</i>	94	<i>P42212</i>	95	<i>P4322</i>	96	<i>P43212</i>
97	<i>I422</i>	98	<i>I4122</i>	99	<i>P4mm</i>	100	<i>P4bm</i>
101	<i>P42cm</i>	102	<i>P42nm</i>	103	<i>P4cc</i>	104	<i>P4nc</i>
105	<i>P42mc</i>	106	<i>P42bc</i>	107	<i>I4mm</i>	108	<i>I4cm</i>
109	<i>I41md</i>	110	<i>I41cd</i>	111	<i>P-42m</i>	112	<i>P-42c</i>
113	<i>P-421m</i>	114	<i>P-421c</i>	115	<i>P-4m2</i>	116	<i>P-4c2</i>
117	<i>P-4b2</i>	118	<i>P-4n2</i>	119	<i>I-4m2</i>	120	<i>I-4c2</i>
121	<i>I-42m</i>	122	<i>I-42d</i>	123	<i>P4/mmm</i>	124	<i>P4/mcc</i>
125	<i>P4/nbm</i>	126	<i>P4/nnc</i>	127	<i>P4/mbm</i>	128	<i>P4/mnc</i>
129	<i>P4/nmm</i>	130	<i>P4/ncc</i>	131	<i>P42/mmc</i>	132	<i>P42/mcm</i>
133	<i>P42/nbc</i>	134	<i>P42/nnm</i>	135	<i>P42/mbc</i>	136	<i>P42/mnm</i>
137	<i>P42/nmc</i>	138	<i>P42/ncm</i>	139	<i>I4/mmm</i>	140	<i>I4/mcm</i>
141	<i>I41/amd</i>	142	<i>I41/acd</i>	143	<i>P3</i>	144	<i>P31</i>
145	<i>P32</i>	146	<i>R3</i>	147	<i>P-3</i>	148	<i>R-3</i>
149	<i>P312</i>	150	<i>P321</i>	151	<i>P3112</i>	152	<i>P3121</i>
153	<i>P3212</i>	154	<i>P3221</i>	155	<i>R32</i>	156	<i>P3m1</i>
157	<i>P31m</i>	158	<i>P3c1</i>	159	<i>P31c</i>	160	<i>R3m</i>
161	<i>R3c</i>	162	<i>P-31m</i>	163	<i>P-31c</i>	164	<i>P-3m1</i>
165	<i>P-3c1</i>	166	<i>R-3m</i>	167	<i>R-3c</i>	168	<i>P6</i>
169	<i>P61</i>	170	<i>P65</i>	171	<i>P62</i>	172	<i>P64</i>
173	<i>P63</i>	174	<i>P-6</i>	175	<i>P6/m</i>	176	<i>P63/m</i>
177	<i>P622</i>	178	<i>P6122</i>	179	<i>P6522</i>	180	<i>P6222</i>
181	<i>P6422</i>	182	<i>P6322</i>	183	<i>P6mm</i>	184	<i>P6cc</i>
185	<i>P63cm</i>	186	<i>P63mc</i>	187	<i>P-6m2</i>	188	<i>P-6c2</i>
189	<i>P-62m</i>	190	<i>P-62c</i>	191	<i>P6/mmm</i>	192	<i>P6/mcc</i>
193	<i>P63/mcm</i>	194	<i>P63/mmc</i>	195	<i>P23</i>	196	<i>F23</i>
197	<i>I23</i>	198	<i>P213</i>	199	<i>I213</i>	200	<i>Pm-3</i>
201	<i>Pn-3</i>	202	<i>Fm-3</i>	203	<i>Fd-3</i>	204	<i>Im-3</i>
205	<i>Pa-3</i>	206	<i>Ia-3</i>	207	<i>P432</i>	208	<i>P4232</i>
209	<i>F432</i>	210	<i>F4132</i>	211	<i>I432</i>	212	<i>P4332</i>
213	<i>P4132</i>	214	<i>I4132</i>	215	<i>P-43m</i>	216	<i>F-43m</i>
217	<i>I-43m</i>	218	<i>P-43n</i>	219	<i>F-43c</i>	220	<i>I-43d</i>
221	<i>Pm-3m</i>	222	<i>Pn-3n</i>	223	<i>Pm-3n</i>	224	<i>Pn-3m</i>
225	<i>Fm-3m</i>	226	<i>Fm-3c</i>	227	<i>Fd-3m</i>	228	<i>Fd-3c</i>
229	<i>Im-3m</i>	230	<i>Ia-3d</i>				

*括号里是在USPEX代码里使用的非标准空间群标号。

9.5 平面群列表

Number	Group
1	p1
2	p2
3	pm
4	pg
5	cm
6	pmm
7	pmg
8	pgg
9	cmm
10	p4
11	p4m
12	p4g
13	p3
14	p3m1
15	p31m
16	p6
17	p6m

9.6 点群列表

罗列出了在Schönflies和Hermann-Maugin（国际）符号中的所有晶体和最重要非晶体点群。

点群列表:

Hermann-Maugin	Schönflies	In USPEX
1	C ₁	C1 or E
2	C ₂	C2
222	D ₂	D2
4	C ₄	C4
3	C ₃	C3
6	C ₆	C6
23	T	T
$\bar{1}$	S ₂	S2
M	C _{1h}	Ch1
mm2	C _{2v}	Cv2
$\bar{2}$	S ₄	S4
$\bar{3}$	S ₆	S6
$\bar{6}$	C _{3h}	Ch3
m $\bar{3}$	T _h	Th
2/m	C _{2h}	Ch2
mmm	D _{2h}	Dh2
4/m	C _{4h}	Ch4
32	D ₃	D3
6/m	C _{6h}	Ch6
432	O	O
422	D ₄	D4
3m	C _{3v}	Cv3
622	D ₆	D6
$\bar{4}3m$	T _d	Td
4mm	C _{4v}	Cv4
$\bar{3}m$	D _{3d}	Dd3
6mm	C _{6v}	Cv6
m $\bar{3}m$	O _h	Oh
$\bar{4}2m$	D _{2d}	Dd2
$\bar{6}2m$	D _{3h}	Dh3
4/mmm	D _{4h}	Dh4
6/mmm	D _{6h}	Dh6
m $\bar{3}m$	O _h	Oh

Important non-crystallographic point groups

Hermann-Maugin	Schönflies	In USPEX
5	C ₅	C5
5/m	S ₅	S5
$\bar{5}$	S ₁₀	S10
5m	C _{v5v}	Cv5
$\bar{10}$	Ch _{5h}	Ch5
52	D ₅	D5
$\bar{5}m$	D _{5d}	Dd5
$\bar{10}2m$	D _{5h}	Dh5
532	I	I
$\bar{5}3m$	I _h	Ih

9.7 USPEX中使用的共价原子半径表

USPEX中所使用的原子半径(单位Å) (主要用于计算硬度*etc.*) :

Z	Element	radius	Z	Element	radius	Z	Element	radius
1	H	0.31	30	Zn	1.22	63	Eu	1.98
2	He	0.28	31	Ga	1.22	64	Gd	1.96
3	Li	1.28	32	Ge	1.20	65	Tb	1.94
4	Be	0.96	33	As	1.19	66	Dy	1.92
5	B	0.84	34	Se	1.20	67	Ho	1.92
6	Csp ³	0.76	35	Br	1.20	68	Er	1.89
	Csp ²	0.73	36	Kr	1.16	69	Tm	1.90
	Csp	0.69	37	Rb	2.20	70	Yb	1.87
7	N	0.71	38	Sr	1.95	71	Lu	1.87
8	O	0.66	39	Y	1.90	72	Hf	1.75
9	F	0.57	40	Zr	1.75	73	Ta	1.70
10	Ne	0.58	41	Nb	1.64	74	W	1.62
11	Na	1.66	42	Mo	1.54	75	Re	1.51
12	Mg	1.41	43	Tc	1.47	76	Os	1.44
13	Al	1.21	44	Ru	1.46	77	Ir	1.41
14	Si	1.11	45	Rh	1.42	78	Pt	1.36
15	P	1.07	46	Pd	1.39	79	Au	1.36
16	S	1.05	47	Ag	1.45	80	Hg	1.32
17	Cl	1.02	48	Cd	1.44	81	Tl	1.45
18	Ar	1.06	49	In	1.42	82	Pb	1.46
19	K	2.03	50	Sn	1.39	83	Bi	1.48
20	Ca	1.76	51	Sb	1.39	84	Po	1.40
21	Sc	1.70	52	Te	1.38	85	At	1.50
22	Ti	1.60	53	I	1.39	86	Rn	1.50
23	V	1.53	54	Xe	1.40	87	Fr	2.60
24	Cr	1.39	55	Cs	2.44	88	Ra	2.21
25	Mn l.s.	1.39	56	Ba	2.15	89	Ac	2.15
	h.s	1.61	57	La	2.07	90	Th	2.06
26	Fe l.s	1.32	58	Ce	2.04	91	Pa	2.00
	h.s.	1.52	59	Pr	2.03	92	U	1.96
27	Co l.s.	1.26	60	Nd	2.01	93	Np	1.90
	h.s.	1.50	61	Pm	1.99	94	Pu	1.87
28	Ni	1.24	62	Sm	1.98	95	Am	1.80
29	Cu	1.32				96	Cm	1.69

Source: Cordero *et al.*, Dalton Trans. 2832-2838, 2008³⁴.

9.8 USPEX使用的默认化合价表

USPEX中使用的化合价valences表（主要用于硬度计算*etc.*）：

Z	Element	valence	Z	Element	valence	Z	Element	valence
1	H	1	35	Br	1	69	Tm	3
2	He	0.5	36	Kr	0.5	70	Yb	3
3	Li	1	37	Rb	1	71	Lu	3
4	Be	2	38	Sr	2	72	Hf	4
5	B	3	39	Y	3	73	Ta	5
6	C	4	40	Zr	4	74	W	4
7	N	3	41	Nb	5	75	Re	4
8	O	2	42	Mo	4	76	Os	4
9	F	1	43	Tc	4	77	Ir	4
10	Ne	0.5	44	Ru	4	78	Pt	4
11	Na	1	45	Rh	4	79	Au	1
12	Mg	2	46	Pd	4	80	Hg	2
13	Al	3	47	Ag	1	81	Tl	3
14	Si	4	48	Cd	2	82	Pb	4
15	P	3	49	In	3	83	Bi	3
16	S	2	50	Sn	4	84	Po	2
17	Cl	1	51	Sb	3	85	At	1
18	Ar	0.5	52	Te	2	86	Rn	0.5
19	K	1	53	I	1	87	Fr	1
20	Ca	2	54	Xe	0.5	88	Ra	2
21	Sc	3	55	Cs	1	89	Ac	3
22	Ti	4	56	Ba	2	90	Th	4
23	V	4	57	La	3	91	Pa	4
24	Cr	3	58	Ce	4	92	U	4
25	Mn	4	59	Pr	3	93	Np	4
26	Fe	3	60	Nd	3	94	Pu	4
27	Co	3	61	Pm	3	95	Am	4
28	Ni	2	62	Sm	3	96	Cm	4
29	Cu	2	63	Eu	3	97	Bk	4
30	Zn	2	64	Gd	3	98	Cf	4
31	Ga	3	65	Tb	3	99	Es	4
32	Ge	4	66	Dy	3	100	FM	4
33	As	3	67	Ho	3	101	Md	4
34	Se	2	68	Er	3	102	No	4

9.9 USPEX使用的默认goodBonds表

USPEX中使用的goodBonds表，主要用于硬度计算*etc.*:

Z	Element	goodBonds	Z	Element	goodBonds	Z	Element	goodBonds
1	H	0.20	35	Br	0.10	69	Tm	0.20
2	He	0.05	36	Kr	0.05	70	Yb	0.20
3	Li	0.10	37	Rb	0.05	71	Lu	0.20
4	Be	0.20	38	Sr	0.10	72	Hf	0.30
5	B	0.30	39	Y	0.20	73	Ta	0.40
6	C	0.50	40	Zr	0.30	74	W	0.30
7	N	0.50	41	Nb	0.35	75	Re	0.30
8	O	0.30	42	Mo	0.30	76	Os	0.30
9	F	0.10	43	Tc	0.30	77	Ir	0.30
10	Ne	0.05	44	Ru	0.30	78	Pt	0.30
11	Na	0.05	45	Rh	0.30	79	Au	0.05
12	Mg	0.10	46	Pd	0.30	80	Hg	0.10
13	Al	0.20	47	Ag	0.05	81	Tl	0.20
14	Si	0.30	48	Cd	0.10	82	Pb	0.30
15	P	0.30	49	In	0.20	83	Bi	0.20
16	S	0.20	50	Sn	0.30	84	Po	0.20
17	Cl	0.10	51	Sb	0.20	85	At	0.10
18	Ar	0.05	52	Te	0.20	86	Rn	0.05
19	K	0.05	53	I	0.10	87	Fr	0.05
20	Ca	0.10	54	Xe	0.05	88	Ra	0.10
21	Sc	0.20	55	Cs	0.05	89	Ac	0.20
22	Ti	0.30	56	Ba	0.10	90	Th	0.30
23	V	0.30	57	La	0.20	91	Pa	0.30
24	Cr	0.25	58	Ce	0.30	92	U	0.30
25	Mn	0.30	59	Pr	0.20	93	Np	0.30
26	Fe	0.25	60	Nd	0.20	94	Pu	0.30
27	Co	0.25	61	Pm	0.20	95	Am	0.30
28	Ni	0.15	62	Sm	0.20	96	Cm	0.30
29	Cu	0.10	63	Eu	0.20	97	Bk	0.30
30	Zn	0.10	64	Gd	0.20	98	Cf	0.30
31	Ga	0.25	65	Tb	0.20	99	Es	0.30
32	Ge	0.50	66	Dy	0.20	100	FM	0.30
33	As	0.35	67	Ho	0.20	101	Md	0.30
34	Se	0.20	68	Er	0.20	102	No	0.30

Bibliography

- [1] J. Maddox. Crystals from first principles. *Nature*, 335:201, 1988.
- [2] A.R. Oganov and C.W. Glass. Crystal structure prediction using ab initio evolutionary techniques: Principles and applications. *The Journal of Chemical Physics*, 124:244704, 2006.
- [3] C.W. Glass, A.R. Oganov, and N. Hansen. USPEX — evolutionary crystal structure prediction. *Comp. Phys. Comm.*, 175:713–720, 2006.
- [4] A.R. Oganov and S. Ono. Theoretical and experimental evidence for a post-perovskite phase of MgSiO₃ in Earth’s D” layer. *Nature*, 430(6998):445–448, July 2004.
- [5] M. Murakami, K. Hirose, K. Kawamura, N. Sata, and Y. Ohishi. Post-perovskite phase transition in MgSiO₃. *Science*, 304(5672):855–858, 2004.
- [6] A.R. Oganov, J.C. Schon, M. Jansen, S.M. Woodley, W.W. Tipton, and R.G. Hennig. *Appendix: First Blind Test of Inorganic Crystal Structure Prediction Methods*, pages 223–231. Wiley-VCH Verlag GmbH & Co. KGaA, 2010.
- [7] C.J. Pickard and R.J. Needs. High-pressure phases of silane. *Phys. Rev. Lett.*, 97:045504, Jul 2006.
- [8] M. Martinez-Canales, A.R. Oganov, Y. Ma, Y. Yan, A.O. Lyakhov, and A. Bergara. Novel structures and superconductivity of silane under pressure. *Phys. Rev. Lett.*, 102:087005, Feb 2009.
- [9] Y. Ma, A.R. Oganov, Y. Xie, Z. Li, and J. Kotakoski. Novel high pressure structures of polymeric nitrogen. *Phys. Rev. Lett.*, 102:065501, 2009.
- [10] C.J. Pickard and R.J. Needs. High-pressure phases of nitrogen. *Phys. Rev. Lett.*, 102:125702, Mar 2009.
- [11] G. Gao, A.R. Oganov, P. Li, Z. Li, H. Wang, T. Cui, Y. Ma, A. Bergara, A.O. Lyakhov, T. Iitaka, and G. Zou. High-pressure crystal structures and superconductivity of stannane (SnH₄). *Proceedings of the National Academy of Sciences*, 107(4):1317–1320, 2010.
- [12] C.J. Pickard and R.J. Needs. Structures at high pressure from random searching. *physica status solidi (b)*, 246(3):536–540, 2009.
- [13] A.O. Lyakhov, A.R. Oganov, H.T. Stokes, and Q. Zhu. New developments in evolutionary structure prediction algorithm USPEX. *Comp. Phys. Comm.*, 184:1172–1182, 2013.
- [14] G.R. Qian, X. Dong, X.-F. Zhou, Y. Tian, A.R. Oganov, and H.-T. Wang. Variable cell nudged elastic band method for studying solid-solid structural phase transitions. *Computer Physics Communications*, 184(9):2111–2118, 2013.
- [15] C. Dellago, P.G. Bolhuis, F.S. Csajka, and D. Chandler. Transition path sampling and the calculation of rate constants. *The Journal of Chemical Physics*, 108(5):1964–1977, 1998.

- [16] S.E. Boulfelfel, A.R. Oganov, and S. Leoni. Understanding the nature of "superhard graphite". *Scientific Reports*, 2(471):1–9, 2012.
- [17] A.R. Oganov, A.O. Lyakhov, and M. Valle. How evolutionary crystal structure prediction works — and why. *Accounts of Chemical Research*, 44(3):227–237, 2011.
- [18] A.R. Oganov and M. Valle. How to quantify energy landscapes of solids. *The Journal of Chemical Physics*, 130:104504, 2009.
- [19] X.-Q. Chen, H. Niu, D. Li, and Y. Li. Modeling hardness of polycrystalline materials and bulk metallic glasses. *Intermetallics*, 19(9):1275–1281, 2011.
- [20] A.R. Oganov and A.O. Lyakhov. Towards the theory of hardness of materials. *Journal of Superhard Materials*, 32(3):143–147, 2010.
- [21] L.S. Dubrovinsky, N.A. Dubrovinskaia, V. Swamy, J. Muscat, N.M. Harrison, R. Ahuja, B. Holm, and B. Johansson. Materials science: The hardest known oxide. *Nature*, 410(6829):653–654, 2001.
- [22] R. Martoňák, A. Laio, M. Bernasconi, C. Ceriani, P. Raiteri, F. Zipoli, and M. Parrinello. Simulation of structural phase transitions by metadynamics. *Z. Krist.*, 220:489–498, 2005.
- [23] A.R. Oganov and C.W. Glass. Evolutionary crystal structure prediction as a tool in materials design. *Journal of Physics: Condensed Matter*, 20(6):064210, 2008.
- [24] A.O. Lyakhov, A.R. Oganov, and M. Valle. *Crystal Structure Prediction Using Evolutionary Approach*, pages 147–180. Wiley-VCH Verlag GmbH & Co. KGaA, 2010.
- [25] Q. Zhu, L. Li, A.R. Oganov, and P.B. Allen. Evolutionary method for predicting surface reconstructions with variable stoichiometry. *Phys. Rev. B*, 87:195317, May 2013.
- [26] W. Zhang, A.R. Oganov, A.F. Goncharov, Q. Zhu, S.E. Boulfelfel, A.O. Lyakhov, E. Stavrou, M. Somayazulu, V.B. Prakapenka, and Z. Konopkova. Unexpected stable stoichiometries of sodium chlorides. *Science*, 342(6165):1502–1505, 2013.
- [27] S.T. Call, D.Yu. Zubarev, and A.I. Boldyrev. Global minimum structure searches via particle swarm optimization. *Journal of Computational Chemistry*, 28(7):1177–1186, 2007.
- [28] Y. Wang, J. Lv, L. Zhu, and Y. Ma. Crystal structure prediction via particle-swarm optimization. *Phys. Rev. B*, 82:094116, Sep 2010.
- [29] G. Mills, H. Jónsson, and G.K. Schenter. Reversible work transition state theory: application to dissociative adsorption of hydrogen. *Surf. Sci.*, 324(2):305–337, 1995.
- [30] G. Henkelman, B.P. Uberuaga, and H. Jónsson. A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *J. Chem. Phys.*, 113(22):9901–9904, 2000.
- [31] G. Henkelman and H. Jónsson. Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points. *J. Chem. Phys.*, 113(22):9978–9985, 2000.

-
- [32] E. Bitzek, P. Koskinen, F. Gähler, M. Moseler, and P. Gumbsch. Structural relaxation made simple. *Phys. Rev. Lett.*, 97(17):170201, 2006.
- [33] M. Valle. STM3: a chemistry visualization platform. *Z. Krist.*, 220:585–588, 2005.
- [34] B. Cordero, V. Gomez, A.E. Platero-Prats, M. Reves, J. Echeverria, E. Cremades, F. Barragan, and S. Alvarez. Covalent radii revisited. *Dalton Trans.*, 21:2832–2838, 2008.