

---

## 摘 要

本文我们提出了一种新的混合方法, 用于从从头算数据点构建光滑的势能面 (PES). 通过接连进行希尔伯特空间再生核 (RKHS) 插值和立方样条插值, 这个方法可以用来有效地构造准确的势能面. 该方法除了可以继承纯再生核方法的大部分优良特性, 比如准确的交换对称性和渐近特性的描写以及可以推导解析导数, 还具有下列优势: 第一, 它的启动时间很短; 当提供的从头算点数非常多时, 启动时间会成为影响拟合效率的重要因素; 过长的启动时间可能导致拟合过程代价极大以至于不可用. 第二, 它具有更好的数值稳定性和准确性. 这是由于该方法减少了再生核插值在求解线性反问题时的病态性.

除此之外, 本文还提出了混合方法的一种快速算法, 这样用于计算势能面及其导数的时间可以大大减少. 最终的计算时间可以和纯再生核方法保持在同一数量级. 为了测试, 我们将此方法应用于  $\text{N}(^2\text{D})+\text{H}_2\rightarrow\text{NH}+\text{H}$  反应体系. 我们从两万多从头算数据点, 构建了  $\text{NH}_2(X^2A')$  态的解析势能面. 基于该势能面, 我们进行了精确量子散射动力学计算. 这些计算展现了新混合插值方法的准确性以及高效性.

**关键词:** 势能面, 量子反应动力学, 插值, 波包

---

## ABSTRACT

A new hybrid interpolation method for constructing smooth potential energy surfaces (PESs) from *ab initio* data is proposed. By successively performing Reproducing Kernel Hilbert Space (RKHS) interpolation and cubic spline interpolation, this method efficiently yields globally accurate potential energy surfaces. In addition to inheriting most advantages of the pure RKHS method, such as correct permutation symmetry and asymptotic behavior, and the capability of providing analytic derivatives, this method offers the following extra benefits: First, a faster initiation time, which would become an important factor to affect the fitting efficiency when the number of available *ab initio* points is large. Lengthy initiation time may makes the fitting process prohibitively expensive; Second, enhanced stability and accuracy due to the decreased ill-conditionality of linear inverse problem when performing RKHS interpolation.

We also proposed a fast algorithm of this hybrid method so that the computational time for the calculation of the PES and its derivatives is greatly reduced and competitive with the pure RKHS method. We have applied this method to  $\text{N}(^2\text{D}) + \text{H}_2 \rightarrow \text{NH} + \text{H}$  reaction to test our method. An adiabatic PES of  $\text{NH}_2(X^2A'')$  state has been constructed from more than 20 000 *ab initio* points. Accurate quantum dynamics results calculated on the constructed PES demonstrates high accuracy and efficiency of this new fast hybrid interpolation method.

**Keywords:** Potential energy surface, Quantum reaction dynamics, Interpolation, Wave packet

---

# 目 录

摘要	i
<b>ABSTRACT</b>	<b>ii</b>
第一章 绪论	<b>1</b>
第二章 方法	<b>3</b>
2.1 混合方法 .....	3
2.2 一维立方样条插值公式 .....	5
2.3 快速混合方法 .....	8
2.4 量子波包计算 .....	11
第三章 结果和讨论	<b>13</b>
3.1 $\text{NH}_2$ 体系势能面的构建 .....	13
3.2 量子波包计算结果 .....	18
第四章 结论	<b>20</b>
参考文献	<b>21</b>

---

## Contents

<b>ABSTRACT</b>	<b>ii</b>
<b>Chapter 1 Introduction</b>	<b>1</b>
<b>Chapter 2 Methods</b>	<b>3</b>
2.1 Hybrid Scheme .....	3
2.2 Formulation of one dimensional cubic spline interpolation .....	5
2.3 Fast Hybrid Method .....	8
2.4 Quantum Wave Packet Calculation .....	11
<b>Chapter 3 Results and Discussions</b>	<b>13</b>
3.1 Construction of the PES of NH <sub>2</sub> System .....	13
3.2 Results of Quantum Wave Packet Calculation .....	18
<b>Chapter 4 Conclusions</b>	<b>20</b>
<b>REFERENCES</b>	<b>21</b>

---

## 第一章 绪论

利用从头算数据点构建全解析势能面 (PES) 是完成分子动力学理论模拟的关键一步. 直接的动力学方法通过实时计算 PES 值, 虽然能避免拟合过程, 但目前从计算的角度来看仍是不切实际的. 在过去的几十年间, 对于设计有效的拟合方法, 人们进行了大量的研究. [1-4] 传统的方法大多是基于对离散的从头算数据点进行最小二乘拟合, 使用的拟合函数是针对特定系统的经验的函数形式. 对于不同系统需要分别分析, 这样的特性限制了对这一类方法的一般化. 除此之外, 通常对于所感兴趣的系统, 设计一个合适的函数形式并不是一件简单的工作. 因此基于通用算法的方法最近一直被广泛研究, 并且在这个方向的工作也有了非常引人注目的进步. 这些基于通用算法的方法包括: 立方样条, [5-7] 最小二乘移动插值, [8-10] 神经网络, [11-15] 改良的 Shepard 插值, [16] 希尔伯特空间再生核, [17] 以及交换不变多项式方法 [3, 15, 18] 等等.

可能其中最流行的方法是立方样条方法, [5-7] 当给定大量的从头算数据点时, 它能产生一个准确的势能面. 但当可获得的从头算数据点比较稀疏时, 产生的势能面通常会带有一些奇异的震荡特性. 由于需要大量的从头算数据点, 这个方法很难用于大于三维的系统. 最近, Ho 等提出了 [17] 一个新的插值方法. 这个方法基于希尔伯特空间再生核 (RKHS) 理论. RKHS 方法已经被证明可以构造高质量的势能面, 并具有显著优势, [17, 19, 20] 比如, 可以确保分子系统的准确的对称性和渐近行为, 以及可以得到势能面的解析导数. 提出这个方法的作者还提出了一个 RKHS 方法的快速算法, 从而 PES 的计算和导数的计算都可以被显著加速. [21] (在后面我们将称这个快速版本为快速 RKHS.) 因此, 使用这种方法构建的势能面既适用于量子动力学计算, 也适用于经典轨线计算. 但是, 当提供大量从头算点时, 这个方法的恰当实现将需要极大的技巧性, 原因是在这种情况下, 与之相关的线性反问题的数值不稳定性会增加. 于是, 我们需要多次调整一些拟合参数直到获得满意的结果. 对于每一次新选择的参数集, 都需要大量的启动时间来求解大规模线性方程组.

---

为了减轻这个问题, 我们提出了一个新的混合方法用于利用从头算数据点构建光滑的势能面. 通过分别在分子构形空间的不同维度应用 RKHS 插值和立方样条插值, 原先求解大规模病态矩阵线性方程组的问题, 将被转化为求解多个小规模良态矩阵线性方程组的问题. 直接实现这种混合方法将会造成程序在计算 PES 值时非常慢, 并且计算成本还会随着从头算点数的增加而急剧增加. 通过组合快速 RKHS [21] 的插值公式和立方样条的插值公式, 我们还提出了混合方法的一种快速算法. 在这个改进的基础上, 基于构建的 PES 进行势能值的计算和导数的计算都会和快速 RKHS 方法一样高效, 并且, 计算代价是独立于从头算数据点的个数的. 为了测试混合方法, 我们使用这个方法基于两万多高理论水平从头算点, 构建了  $\text{NH}_2$  体系的全局从头算势能面. 基于构建的势能面我们进行了准确的量子动力学计算.

本论文主要按以下方式组织. 在第二章我们将描述混合方法的理论细节, 包括方法的直接实现和快速算法. 在第二章我们还会简要概述所使用的量子波包方法. 计算结果将在第三章给出. 第四章是小结和结论.

---

## 第二章 方法

为了定量比较不同的算法, 此处我们使用大 O 标记来刻画时间复杂度和空间复杂度, [22] 这种标记在计算机科学中被广泛应用, 用于估计算法的效率.

### 2.1 混合方法

根据 Ho 等人的工作, [17] 多维势能面的 RKHS 插值可以形式地写为 [17]

$$V^+(\mathbf{R}) = \sum_{\kappa=1}^M \beta_{\kappa} Q(\mathbf{R}_{\kappa}, \mathbf{R}) \quad (2.1)$$

其中  $V^+(\mathbf{R})$  表示从一个给定的在  $M$  个核构形  $\mathbf{R} = \mathbf{R}_{\kappa}$  的从头算能量集

$$\mathcal{V}(\mathbf{R}_{\kappa}), \quad \kappa = 1, 2, \dots, M \quad (2.2)$$

构建的势能面. 此处  $N$  体系统的  $3N-6$  个独立的内坐标合并表示为  $\mathbf{R}$ .  $Q(\mathbf{R}_{\kappa}, \mathbf{R})$  表示再生核. 展开系数  $\beta_{\kappa}$  是通过求解下面的线性方程组来得到的

$$\sum_{\lambda=1}^M Q_{\kappa\lambda} \beta_{\lambda} = \mathcal{V}(\mathbf{R}_{\kappa}), \kappa = 1, 2, \dots, M \quad (2.3)$$

其中  $Q_{\kappa\lambda} = Q(\mathbf{R}_{\kappa}, \mathbf{R}_{\lambda})$ . 只要对称正定  $M \times M$  矩阵  $\mathbf{Q}_M = [Q_{\kappa\lambda}]$  是良态的, 可以直接数值求解这个线性方程组. 但是, 若从头算点的点数  $M$  很大, 矩阵将会出现秩亏损, 这样按照通常方法就很难求得线性方程组的解. 在这种情况下, 我们需要使用正规化方法.

最初我们使用 RKHS 方法从两万多个从头算数据点构建  $\text{NH}_2$  系统解析势能面的尝试并不成功, 即使使用了正规化方法也是如此. 构建的势能面对于正规化参数的取值非常敏感, 另外, 由于从头算数据点数量众多, 启动过程非常耗时. 过长的启动时间使得我们难以尝试很多次来寻找最合适的正规化参数值. 这种困难使得我们开始考虑提出一种新的称为混合方法的拟合方法. 考虑到应用 RKHS 方法的困难其实在于过多的从头算数据点, 我们把完整的  $3N-6$  维分子构形空间分割为两个或多个子空间并采用不同的方法处理各子空间. 我们将 RKHS 插值应

用于其中一个子空间, 而对其他子空间则采取其他的拟合方法, 比如立方样条方法. 这么做由于减少了维度以及与之相关的线性方程组关联的矩阵尺寸, 可以预期对于选择的子空间 RKHS 方法将工作良好.

全维构型空间可以被分割为两个或多个子空间, 我们可以有很多方法处理每个子空间. 为了展现混合方法的更多细节, 我们将着重讨论一种特定的混合方式, 在这种方式里完整的  $d = 3N - 6$  维构型空间被划分为两个子空间, 其中一个是一维的, 另一个是  $d - 1$  维的. 并且, 我们对前者应用立方样条插值方法, 而对后者应用 RKHS 插值方法. 为方便起见, 我们假设  $M$  个从头算点是服从一个格点分布的. 但是, 应当指出混合方法并非被限制于这一种特殊的情形. 混合方法同样可以应用于非格点分布的从头算点的情形, 这是由于首先 RKHS 方法本身是适用于这种情形的, 这在文献 [17] 和 [21] 中已经得到说明. 其次, 正如即将在本文后面进行说明的, 混合方法所做的是通过完成第一维每个格点关联的  $d - 1$  维 RKHS 插值得到该维度的势能值, 用以完成一维立方样条拟合. 我们的测试显示包含少数非格点分布的点仅仅对拟合效率有非常小的影响. 尽管如此, 完全随机分布的从头算数据点将使得算法非常低效. 在格点分布的假设下, 有以下等式:

$$M = m^{(1)}m^{(2)} \dots m^{(d)} \quad (2.4)$$

其中  $m^{(i)}$  表示在第  $i$  维选择的坐标数.

我们在第一维的每个格点总计进行  $m^{(1)}$  个独立的  $d - 1$  维 RKHS 插值. 为了得到全维势能面,  $m^{(i)}$  个构建的  $d - 1$  维势能面将由立方样条插值拟合为一个势能面. 例如, 为了在一个任意选择的构形计算势能值, 比如选择的构形是  $\mathbf{R} = (x^{(1)}, x^{(2)}, \dots, x^{(d)})$ , 需要先计算  $m^{(1)}$  个  $d - 1$  维势能面, 这些势能面其实对应于把第一个坐标  $x^{(1)}$  设定为第一维的格点值  $x_i^{(1)}$  时的全维势能面, 即

$$V_i = V^+(x_i^{(1)}, x^{(2)}, \dots, x^{(d)}), \quad i = 1, 2, \dots, m^{(1)}. \quad (2.5)$$

在这之后, 最终的在构形  $\mathbf{R} = (x^{(1)}, x^{(2)}, \dots, x^{(d)})$  处的全维势能值就可以通过一维立方样条插值来得到.



求解一个对应于  $M \times M$  矩阵的线性方程组的时间和空间复杂度分别是  $O(M^3)$  和  $O(M^2)$ . [23] 混合方法需要求解  $m^{(1)}$  个规模被减小的矩阵对应的方程组. 因此, 对于启动过程时间和空间复杂度分别是  $O(m^{(1)}(m^{(2)})^3 \dots (m^{(d)})^3)$  和  $O(m^{(1)}(m^{(2)})^2 \dots (m^{(d)})^2)$ . 与纯 RKHS 方法相比, 启动过程的时间和空间复杂度分别减小了  $(m^{(1)})^2$  和  $m^{(1)}$  因子. 当可得到的从头算数据点很多时, 这种减少是非常重要的, 因为此时求解线性方程组会变得非常耗时. 当然考虑到最初为了确定对应的系数而需要求解的线性方程组可以一次解出, 而系数可以永久保存, 即这个非常耗时的过程仅仅与启动时间相关联. 但是, 当从头算数据很大时, 构建的势能面对于正规化系数是敏感的. 从而启动过程就会变得代价极高, 因为我们需要反复求解线性方程组来确定正规化系数的最合适值.

除了启动时间, 另一个势能面拟合的重要问题是我们要求构建的势能面应该能够较高效地算出势能值及其导数. 直接实现纯 RKHS 方法 [17] 并不能满足这个要求. 为了解决这个问题, Hollebeek 等 [21] 提出了一个 RKHS 方法的快速算法, 通过预先计算相关的系数, 可以让势能面求值过程的计算时间与从头算数据的点数无关. 这个快速算法的时间复杂度是  $O(\Sigma(\log_2 m^{(j)}))$ , 这是把二分搜索考虑为其中最耗时的步骤. 对于上面提到的混合方法而言, 由于在立方样条插值中求解二阶导数必须对 (2.5) 式中每一个  $V_i$  重复, PES 计算的时间复杂度是  $O((m^{(1)})^2)$ , 这个复杂度比快速 RKHS 方法要大. 幸运的是, 混合方法可以以一种更有效的方式实现, 这种实现方式将减少 PES 计算的时间复杂度, 使之可以和快速 RKHS 方法达到同一数量级, 而与此同时启动过程的时间复杂度将只有细微的改变. 这个快速算法的核心思想是, 将两个插值方法 RKHS 和立方样条的插值公式进行组合, 从而它们的相关系数可以合并. 合并的系数可以预先计算并存储以重复利用. 为实现这一目的, 我们需要首先推导立方样条插值的数学公式.

## 2.2 一维立方样条插值公式

给定一个离散函数  $V_i = V(x_i)$  ( $i = 1, 2, 3, \dots, N$ ), 其中  $x_1 < x_2 < \dots < x_N$ , 立方样条插值将给出函数  $V(x)$  在任意坐标区间, 比如  $x_\alpha < x < x_{\alpha+1}$  ( $\alpha =$

$1, 2, \dots, N-1$ ) 的近似值. 样条插值使用下列公式 [24]

$$V(x) = A_\alpha(x)V_\alpha + B_\alpha(x)V_{\alpha+1} + C_\alpha(x)V''_\alpha + D_\alpha(x)V''_{\alpha+1} \quad (2.6)$$

其中系数  $A_\alpha(x), B_\alpha(x), C_\alpha(x)$  和  $D_\alpha(x)$  都是  $x$  的多项式函数

$$\begin{aligned} A_\alpha(x) &= \frac{x_{\alpha+1} - x}{x_{\alpha+1} - x_\alpha}, \\ B_\alpha(x) &= \frac{x - x_\alpha}{x_{\alpha+1} - x_\alpha}, \\ C_\alpha(x) &= \frac{1}{6}[A_\alpha^3(x) - A_\alpha(x)](x_{\alpha+1} - x_\alpha)^2, \\ D_\alpha(x) &= \frac{1}{6}[B_\alpha^3(x) - B_\alpha(x)](x_{\alpha+1} - x_\alpha)^2, \end{aligned} \quad (2.7)$$

而  $V''_\alpha$  是  $V(x)$  在  $x = x_\alpha$  的二阶导数. 通过强制函数的一阶导数在两个相邻区间连续, 我们可以得到下面关于  $V''_\alpha$  的线性方程组

$$\begin{aligned} \frac{x_\alpha - x_{\alpha-1}}{6}V''_{\alpha-1} + \frac{x_{\alpha+1} - x_{\alpha-1}}{3}V''_\alpha + \frac{x_{\alpha+1} - x_\alpha}{6}V''_{\alpha+1} \\ = \frac{V_{\alpha+1} - V_\alpha}{x_{\alpha+1} - x_\alpha} - \frac{V_\alpha - V_{\alpha-1}}{x_\alpha - x_{\alpha-1}}, \text{ for } \alpha = 2, \dots, N-1 \end{aligned} \quad (2.8)$$

但是上述公式对于得到  $V''_\alpha$  的唯一解并不足够. 通常需要提供在两个端点  $x_1$  和  $x_N$  的附加条件. 例如, 对于所谓的自然立方样条, 将要求下面的条件

$$V''_1 = V''_N = 0. \quad (2.9)$$

以更紧凑的形式写出这些方程, 我们有

$$\mathbb{X}\mathbf{V}'' = \mathbb{Y}\mathbf{V} \quad (2.10)$$

其中  $\mathbf{V}$  和  $\mathbf{V}''$  分别是由  $V_\alpha$  和  $V''_\alpha$  组成的  $N \times 1$  矩阵.  $\mathbb{X}$  和  $\mathbb{Y}$  则是  $N \times N$  三对角矩阵.  $\mathbf{V}''$  的解由下式给出

$$\mathbf{V}'' = \mathbb{F}\mathbf{V} \quad (2.11)$$

其中

$$\mathbb{F} = \mathbb{X}^{-1}\mathbb{Y}. \quad (2.12)$$

把 (2.11) 式代入 (2.6) 式, 可以得到立方样条插值的形式表达式

---


$$V(x) = \sum_l \left\{ A_\alpha(x) \delta_{\alpha l} + B_\alpha(x) \delta_{\alpha+1, l} + \sum_s [C_\alpha(x) \delta_{\alpha s} + D_\alpha(x) \delta_{\alpha+1, s}] F_{sl} \right\} V_l \quad (2.13)$$

其中  $F_{sl}$  表示  $\mathbb{F}$  的矩阵元,  $\delta_{ij}$  是克罗内克 delta 函数. 由于系数  $A_\alpha(x), B_\alpha(x), C_\alpha(x)$  和  $D_\alpha(x)$  都是  $x$  的线性或三次函数, 可以进一步将它们表示为

$$\begin{aligned} A_\alpha(x) &= \sum_\lambda A_{\alpha\lambda} \cdot x^\lambda, \\ B_\alpha(x) &= \sum_\lambda B_{\alpha\lambda} \cdot x^\lambda, \\ C_\alpha(x) &= \sum_\lambda C_{\alpha\lambda} \cdot x^\lambda, \\ D_\alpha(x) &= \sum_\lambda D_{\alpha\lambda} \cdot x^\lambda. \end{aligned} \quad (2.14)$$

将这些表达式代入 (2.13) 式, 我们得到立方样条插值的最终数学表达式

$$V(x) = \sum_{l, \lambda} E_{\alpha\lambda l} V_l \cdot x^\lambda \quad (2.15)$$

其中  $E_{\alpha\lambda l}$  由下式给出

$$E_{\alpha\lambda l} = A_{\alpha\lambda} \delta_{\alpha l} + B_{\alpha\lambda} \delta_{\alpha+1, l} + \sum_s [C_{\alpha\lambda} \delta_{\alpha s} + D_{\alpha\lambda} \delta_{\alpha+1, s}] F_{sl}. \quad (2.16)$$

考虑到  $\alpha$  和  $\lambda$  的可能值数目是有限的, 而  $\sum_l E_{\alpha\lambda l} V_l$  在给定  $\alpha$  和  $\lambda$  的值后是独立于  $x$  的, 我们可以遍历所有的  $\alpha$  和  $\lambda$  值来预先计算所有对应的  $\sum_l E_{\alpha\lambda l} V_l$  并存储以在将来使用. 假设使用二分搜索 [24] 方法来定位  $x$ , 计算  $V(x)$  的时间复杂度将是  $O(\log_2 N)$ . 考虑到  $\mathbb{X}$  和  $\mathbb{Y}$  都是三对角矩阵, 计算  $\mathbb{F} = \mathbb{X}^{-1}\mathbb{Y}$  只需要  $O(N^2)$  的算术操作. [23] 除此之外,  $E_{\alpha\lambda l}$  表达式中的  $\delta$  并不需要显式求和. 所以, 预先计算  $E_{\alpha\lambda l}$  的时间复杂度是  $O(N_\alpha N_\lambda N + N^2)$ . 对

于立方样条插值, 有  $N_\alpha = N - 1$  以及  $N_\lambda = 4$ . 因此, 启动过程和 PES 计算的时间复杂度分别是  $O(N^2)$  和  $O(\log_2 N)$ .

### 2.3 快速混合方法

我们从 (2.15) 式出发, 对于一个  $d$  维势能面, 如果我们对第一个自由度使用一维立方样条插值, 势能面可以表示为

$$V(x^{(1)}, x^{(2)}, \dots, x^{(d)}) = \sum_{l, \lambda} E_{\alpha \lambda l} V_l(x^{(2)}, \dots, x^{(d)}) \cdot (x^{(1)})^\lambda. \quad (2.17)$$

在这种情形下, 系数  $V_l$  是一个  $d - 1$  维函数而不是一个数. 这个  $d - 1$  维函数  $V_l(x^{(2)}, \dots, x^{(d)})$  可以由混合方法中的快速 RKHS 部分得到. 根据 Hollebeek 等的文献,  $V_l(x^{(2)}, \dots, x^{(d)})$  可以表达为下列形式 [21]

$$\begin{aligned} V_l(x^{(2)}, \dots, x^{(d)}) &= \sum_{k=1}^{M_1^{(2)}} f_{1k}^{(2)}(x^{(2)}) \gamma_{l1k} + \sum_{k=1}^{M_2^{(2)}} [f_{3k}^{(2)}(x^{(2)}) \gamma_{l2k} + f_{2k}^{(2)}(x^{(2)}) \gamma_{l3k}], \\ &\gamma_{lm^{(2)}k^{(2)} \dots m^{(n)}k^{(n)}} \\ &= \sum_{k=1}^{M_1^{(n+1)}} f_{1k}^{(n+1)}(x^{(n+1)}) \gamma_{lm^{(2)}k^{(2)} \dots m^{(n)}k^{(n)}1k} \\ &+ \sum_{k=1}^{M_2^{(n+1)}} [f_{3k}^{(n+1)}(x^{(n+1)}) \gamma_{lm^{(2)}k^{(2)} \dots m^{(n)}k^{(n)}2k} \quad (2.18) \\ &+ f_{2k}^{(n+1)}(x^{(n+1)}) \gamma_{lm^{(2)}k^{(2)} \dots m^{(n)}k^{(n)}3k}], \\ &\gamma_{lm^{(2)}k^{(2)} \dots m^{(d)}k^{(d)}} = \sigma_{lm^{(2)}k^{(2)} \dots m^{(d)}k^{(d)}}, \\ &\sigma_{lm^{(2)}k^{(2)} \dots m^{(d)}k^{(d)}} \\ &= \sum_{i^{(2)} \in s^{(2)}(m^{(2)})} \dots \sum_{i^{(d)} \in s^{(d)}(m^{(d)})} \beta_{li^{(2)}i^{(3)} \dots i^{(d)}} \\ &\times \prod_{j=2}^d p_{m^{(j)}k^{(j)}}^{(j)} f_{m^{(j)}k^{(j)}}^{(j)}(x_{i^{(j)}}^{(j)}), \end{aligned}$$

$$\begin{aligned}
\sigma_{lm^{(2)}k^{(2)}\dots m^{(d)}k^{(d)}} &= \sum_{i^{(2)} \in S^{(2)}(m^{(2)})} \dots \sum_{i^{(d)} \in S^{(d)}(m^{(d)})} \beta_{li^{(2)}i^{(3)}\dots i^{(d)}} \\
&\times \prod_{j=2}^d p_{m^{(j)}k^{(j)}}^{(j)} f_{m^{(j)}k^{(j)}}^{(j)}(x_{i^{(j)}}^{(j)}), \\
&x_{\alpha^{(i)}}^{(i)} \leq x^{(i)} < x_{\alpha^{(i)}+1}^{(i)}, \\
&p_{3k}^{(j)} = p_{2k}^{(j)},
\end{aligned}$$

其中  $p_{ik}^{(j)}$  和  $f_{ik}^{(j)}$  分别是待定系数和函数的集合, 它们依赖于所选择的第  $j$  维的再生核的类型.  $M_i^{(j)}$  是  $k$  的允许取值的数目,  $\gamma$  是中间系数,  $\sigma$  是可以预先计算的系数.  $\beta_{li^{(2)}i^{(3)}\dots i^{(d)}}$  是求解线性方程组 (2.3) 式得到的展开系数.  $\alpha^{(i)}$  表示第  $i$  维格点的序数.

需要指出的是, 这个  $V_l$  的表达式包括很多计算细节, 因此使用这个形式并不利于后面的讨论. 为了简化起见, 我们引入复合指标  $h^{(j)}$  来代替  $m^{(j)}k^{(j)}$ . 在调整一些  $h^{(j)}$  的顺序之后,  $V_l$  可以表示为

$$\begin{aligned}
V_l(x^{(2)}, \dots, x^{(d)}) &= \sum_{h=1}^{M^{(2)}} f_h^{(2)}(x^{(2)}) \gamma_{lh}, \\
\gamma_{lh^{(2)}\dots h^{(n)}} &= \sum_{h=1}^{M^{(n+1)}} f_h^{(n+1)}(x^{(n+1)}) \gamma_{lh^{(2)}\dots h^{(n)}h}, \\
\gamma_{lh^{(2)}\dots h^{(d)}} &= \sigma_{lh^{(2)}\dots h^{(d)}}.
\end{aligned} \tag{2.19}$$

由于  $\sigma$  可以预先计算, 这里  $\sigma$  的表达式并不重要. 我们需要注意的是  $\sigma$  是  $\alpha^{(j)}$  的函数, 这意味着

$$\sigma_{lh^{(2)}\dots h^{(d)}} = \sigma_{lh^{(2)}\dots h^{(d)}}(\alpha^{(2)}, \dots, \alpha^{(d)}). \tag{2.20}$$

所有的  $\alpha^{(j)}$  都是整数, 它们所有可取的值组成一个有限集, 因此我们可以遍历  $h^{(2)}, h^{(3)}, \dots, h^{(d)}, \alpha^{(2)}, \alpha^{(3)}, \dots, \alpha^{(d)}$  和  $l$  的所有取值来预先计算  $\sigma$ . 省略中间系数  $\gamma$ , 我们可以将  $V_l$  重新写为

$$V_l(x^{(2)}, \dots, x^{(d)}) = \sum_{h^{(2)}=1}^{M^{(2)}} \sum_{h^{(3)}=1}^{M^{(3)}} \dots \sum_{h^{(d)}=1}^{M^{(d)}} \sigma_{lh^{(2)} \dots h^{(d)}}(\alpha^{(2)}, \dots, \alpha^{(d)}) \quad (2.21)$$

$$\times f_h^{(d)}(x^{(d)}) \cdot \dots \cdot f_h^{(3)}(x^{(3)}) \cdot f_h^{(2)}(x^{(2)}).$$

把上述  $V_l$  的表达式代入 (2.17) 式, 我们得到快速混合方法的插值公式

$$V(x^{(1)}, x^{(2)}, \dots, x^{(d)}) = \sum_{\lambda}^{N_{\lambda}} \sum_{h^{(2)}=1}^{M^{(2)}} \sum_{h^{(3)}=1}^{M^{(3)}} \dots \sum_{h^{(d)}=1}^{M^{(d)}} C_{\alpha^{(1)}\alpha^{(2)} \dots \alpha^{(d)} \lambda h^{(2)} \dots h^{(d)}} \quad (2.22)$$

$$\times f_h^{(d)}(x^{(d)}) \cdot \dots \cdot f_h^{(3)}(x^{(3)}) \cdot f_h^{(2)}(x^{(2)}) \cdot (x^{(1)})^{\lambda},$$

其中的组合系数由下式给出

$$C_{\alpha^{(1)}\alpha^{(2)} \dots \alpha^{(d)} \lambda h^{(2)} \dots h^{(d)}} = \sum_l^{m^{(1)}} E_{\alpha^{(1)}\lambda l} \sigma_{lh^{(2)} \dots h^{(d)}}(\alpha^{(2)}, \dots, \alpha^{(d)}). \quad (2.23)$$

基于这个快速混合方法的公式, 许多耗时的重复性算术操作都可以在启动阶段完成, 相关的系数可以预先计算并存储, 从而被重复利用. 启动过程可以分为四步. 第一步是求解线性方程组来得到展开系数  $\beta_{li^{(2)}i^{(3)} \dots i^{(d)}}$ . 这一步需要  $O(m^{(1)}(m^{(2)})^3 \dots (m^{(d)})^3)$  的时间复杂度, 也是最耗时的一步. 第二步是预先计算系数  $\sigma_{lh^{(2)} \dots h^{(d)}}(\alpha^{(2)}, \dots, \alpha^{(d)})$ , 这需要  $O(m^{(1)}m^{(2)} \dots m^{(d)})$  时间复杂度. 第三步是预先计算系数  $E_{\alpha^{(1)}\lambda l}$ , 它的时间复杂度是  $O((m^{(1)})^2)$ . 最后一步是得到组合系数  $C_{\alpha^{(1)}\alpha^{(2)} \dots \alpha^{(d)} \lambda h^{(2)} \dots h^{(d)}}$ , 这需要  $O((m^{(1)})^2 m^{(2)} \dots m^{(d)})$  时间复杂度, 其中考虑了对  $l$  指标的求和. 因此, 快速混合方法启动过程总的时间复杂度是  $O(m^{(1)}(m^{(2)})^3 \dots (m^{(d)})^3)$ , 这比纯 RKHS 方法的启动时间复杂度要小 (减小了一个  $m^{(1)}$  平方因子).

我们发现 (2.22) 式中所有的  $M^{(j)}$  都是很小的常数, 它们依赖于再生核的类型, 而  $N_{\lambda}$  等于 4. 因此二分搜索应该被认为是最耗时的步骤. 因此, 混合方法对于势能面求值具有  $O(\sum(\log_2 m^{(j)}))$  的复杂度, 这几乎和快速 RKHS 方法的求值复杂度相等.

从快速混合方法的公式我们可以看出, 其导数的解析形式也可以很容易地

求出. 由于篇幅所限我们在此省略推导的细节. 导数的解析表达式对于准经典轨线计算是尤其有用的. [25]

## 2.4 量子波包计算

在此我们对动力学计算的方法给出一个简要的概述. 我们使用了高效的切比雪夫实波包传播方法来计算精确全维量子力学反应概率和积分截面 (ICS). [26-31]

使用反应物雅克比坐标  $(R, r, \gamma)$  哈密顿量可以表示为 [32]

$$\hat{H} = -\frac{1}{2\mu_R} \frac{\partial^2}{\partial R^2} - \frac{1}{2\mu_r} \frac{\partial^2}{\partial r^2} + \frac{\hat{j}^2}{2\mu_r r^2} + \frac{\hat{l}^2}{2\mu_R R^2} + V(R, r, \gamma) \quad (2.24)$$

其中  $R$  和  $r$  分别表示 N-H<sub>2</sub> 和 H-H 之间的距离,  $\gamma$  是  $R$  和  $r$  间的夹角.  $\mu_R$  和  $\mu_r$  分别是与  $R$  和  $r$  坐标关联的约化质量.  $V(R, r, \gamma)$  是分子系统的势能面.  $\hat{j}^2$  表示双原子转动角动量平方算符, 而  $\hat{l}^2$  表示轨道角动量平方算符.

设计好的初始波包  $|\psi_0\rangle$  将按照以下三项迭代式从反应物渐近区开始传播

$$|\psi_{k+1}\rangle = D(2\hat{H}_{\text{norm}}|\psi_k\rangle - D|\psi_{k-1}\rangle), \quad k > 1 \quad (2.25)$$

以及

$$|\psi_1\rangle = D\hat{H}_{\text{norm}}|\psi_0\rangle \quad (2.26)$$

其中  $D$  表示放置在  $R$  (和  $r$ ) 格点边缘的吸附函数, 用来满足出射边界条件. 哈密顿量需要进行适当的归一化 ( $\hat{H}_{\text{norm}}$ ) 来避免切比雪夫传播的发散. [32]

为了完成哈密顿量作用到波包 ( $\hat{H}_{\text{norm}}|\psi_k\rangle$ ) 的数值计算, 哈密顿算符和波包必须进行离散化. 我们使用了混合的格点/基组表象来实现这个目的. 简单地说, 对两个径向雅克比坐标我们定义了等距傅里叶格点. 相关联的动能算符的作用是通过快速正弦傅里叶变换实现的. 对于角度自由度, 交替使用了有限基组表象 (FBR) 和离散变量表象 (DVR). [33] FBR 被用来计算转动动能算符的作用. 接

---

着 FBR 通过伪光谱变换被转换成 DVR 来计算势能算符的作用. 通过计算经过产物谷处的通道的流我们可以得到反应概率. [34] 进行部分波的求和不难得到 ICS.



---

## 第三章 结果和讨论

### 3.1 $\text{NH}_2$ 体系势能面的构建

我们使用快速混合方法从 Zhou 等 [35] 的 25 141 个从头算数据点构建了一个新的  $\text{N}(^2\text{D})+\text{H}_2\rightarrow\text{NH}+\text{H}$  反应体系的从头算势能面. 从头算数据点是在一个内坐标的截断直积格点上提供的. 内坐标包括 N-H 和 H-H 距离以及 N-H-H 弯曲角, 表示为  $(R_{\text{NH}_1}, R_{\text{H}_1\text{H}_2}, \xi_{\text{NH}_1\text{H}_2})$ , 其中我们使用标记  $\text{H}_1$  和  $\text{H}_2$  来区分两个氢原子. 在  $R_{\text{NH}_1} \in [1.1a_0, 15a_0]$  定义了 33 个格点, 在  $R_{\text{H}_1\text{H}_2} \in [0.7a_0, 14a_0]$  定义了 31~53 个格点, 而在  $\xi_{\text{NH}_1\text{H}_2} \in [0^\circ, 180^\circ]$  定义了 21 个格点. [35] 除了少数点不在格点上, 大部分数据点都服从格点分布.

多体展开 [36] 曾经在 RKHS [19] 和其他拟合算法中被广泛使用. 混合方法同样也可以结合这一处理方式在多个方面得到提高. 首先, 渐近衰减的三体项一般和其他项相比具有较小的值 [13]. 从而三体项的拟合误差就会减小, 而三体项的拟合误差是总误差的主要来源, 从而总误差也就会减小. 其次, 它可以使得势能面更好地满足渐近性质 [17, 36]. 我们用下面的多体展开式表示势能面

$$\begin{aligned} V_{\text{HHN}}(R_{\text{NH}_1}, R_{\text{NH}_2}, R_{\text{H}_1\text{H}_2}) \\ = V_{\text{NH}}^{(2)}(R_{\text{NH}_1}) + V_{\text{NH}}^{(2)}(R_{\text{NH}_2}) + V_{\text{HH}}^{(2)}(R_{\text{H}_1\text{H}_2}) \\ + V_{\text{HHN}}^{(3)}(R_{\text{NH}_1}, R_{\text{NH}_2}, R_{\text{H}_1\text{H}_2}), \end{aligned} \quad (3.1)$$

其中  $R_{\text{NH}_1}, R_{\text{NH}_2}$  和  $R_{\text{H}_1\text{H}_2}$  表示三个原子核间距.  $V^{(2)}$  和  $V^{(3)}$  分别表示两体项和三体项. 两体项  $V_{\text{NH}}^{(2)}(R_{\text{NH}})$  和  $V_{\text{HH}}^{(2)}(R_{\text{H}_1\text{H}_2})$  使用一维 RKHS 插值构建. 三体项  $V_{\text{HHN}}^{(3)}(R_{\text{NH}_1}, R_{\text{NH}_2}, R_{\text{H}_1\text{H}_2})$  由混合方法构建, 所用到的离散值现在由从头算能量和三个两体项能量之和的差给出

(3.2)

$$\begin{aligned}
V_{HHN}^{(3)}(R_{NH_1}, R_{NH_2}, R_{H_1H_2}) \\
&= V_{HHN}(R_{NH_1}, R_{NH_2}, R_{H_1H_2}) \\
&\quad - \left[ V_{NH}^{(2)}(R_{NH_1}) + V_{NH}^{(2)}(R_{NH_2}) + V_{HH}^{(2)}(R_{H_1H_2}) \right].
\end{aligned}$$

我们在两个径向坐标 ( $R_{NH_1}$  和  $R_{H_1H_2}$ ) 组成的二维构形空间应用 RKHS 插值, 而在角度自由度 ( $\xi_{NH_1H_2}$ ) 关联的空间应用一维立方样条插值.

为了完成 RKHS 插值, 我们从两个一维再生核的直积构建了如下的二维再生核

$$Q(x, x', y, y') = q_1^{2,6}(x, x') q_1^{2,6}(y, y') \quad (3.3)$$

其中一维再生核具有如下形式 [17]

$$q_1^{n,m}(x, x') = n^2 x_{>}^{-(m+1)} B(m+1, n) {}_2F_1\left(-n+1, m+1; n+m+1, \frac{x_{<}}{x_{>}}\right) \quad (3.4)$$

其中  $x_{>}$  和  $x_{<}$  分别是  $x$  和  $x'$  中的较大者和较小者,  $B(m, n)$  是 beta 函数, 而  ${}_2F_1(n_1, n_2; n_3, z)$  是超几何函数.

在对  $\xi_{NH_1H_2}$  自由度应用一维立方样条插值的过程中, 使用了下面的边界条件来替代自然样条条件

$$V'_1 = V'_N = 0, \quad (3.5)$$

这是考虑到系统是关于  $\xi_{NH_1H_2} = 0$  和  $\xi_{NH_1H_2} = \pi$  对称的. 最终势能面  $V_{HHN}(R_{NH_1}, R_{NH_2}, R_{H_1H_2})$  由构建的三体项  $V_{HHN}^{(3)}$  和二体项之和给出.

我们同样尝试从同样的从头算点集使用纯 RKHS 方法构建势能面. 图 1 显示了分别用三种不同方法, 纯 RKHS 方法, 混合方法和三维立方样条方法 (参考文献 [35]) 构建的  $NH_2$  势能面的等高线图 (固定  $\xi_{NH_1H_2} = \pi/2$ ). 如图 1(a) 所示, 纯 RKHS 插值构建的势能面具有许多不好的震荡, 这很显然是求解大尺度线性方程组的不稳定性导致的 (使用了正规化方法 [17]). 势能面的质量对于正规化参数的取值很敏感. 除此之外, 耗时的启动过程使得我们难以多次求解线性方程组.

---

这些问题需要非常好的线性方程求解算法来解决.

在混合方法中, 由于我们在一个低维的子空间进行 RKHS 插值, 与线性方程组相关的矩阵尺寸被显著减少了. RKHS 方法在此种情形下工作良好, 因为更小的矩阵尺寸使得线性系统的病态性降低了. 如图 1(b) 所示, 快速混合方法构建的势能面的等高线图是光滑而且合理的. 势能面的主要特性, 比如小的早期势垒, 深势阱以及渐近区的平整度, 都得到了很好的再现. 它还和三维样条插值方法构建的势能面高度一致. 如表 1 所示, 由混合方法构建的势能面在几个关键点的构形和能量都和之前的工作 [35] 达到了高度一致. 我们还计算了在几个不同上界的混合方法势能面的均方根 (RMS) 误差. 这些误差是由于引入了正规化参数而造成的. 如表 2 所示, RMS 误差都小于 1 千卡/摩尔, 并且当选择更低的能量上界时, 误差还会降低, 这说明大的误差主要是由能量高部分的点导致的. 为了进一步测试方法的准确性, 我们在构建的势能面的基础上进行了精确的三维量子动力学计算, 这部分结果将在第 3.2 节给出.

除了构建的势能面的质量, 另一个重要问题是拟合方法计算势能值及其导数的速度. 在表 3 中我们比较了三种不同拟合方法的启动时间和计算给定构形的势能值的时间. 表格显示尽管纯 RKHS 方法计算势能值很快, 但是它启动很慢. 与之相反, 快速混合方法把启动时间降低到一个合理的水平, 而并没有过多影响势能值计算时间, 势能值计算的速度仍然很快. 除此之外, 在启动阶段与纯 RKHS 方法 (2.45 GB) 相比, 混合方法使用了更少的内存 (24.1 MB).

三维立方样条方法的启动时间几乎可以忽略, 但是它的势能值计算时间特别长, 如果按照一种不高效的方式来实现的话 (我们在表 3 中将这一方法标记为“样条 (慢)”), 也就是接连进行三个一维立方样条插值的方法. [35] 为了对三维立方样条方法做一个公平的评估, 我们研究了三维立方样条插值的一种高效算法 (在表 3 中标记为“样条 (快)”). 这种高效算法的主要思想是把第 2.2 节所述的一维立方样条方法拓展到三维情形. 立方样条方法的快速版本显著提高了势能面计算速度. 在表 3 列出的这三种方法中, 快速立方样条看上去是表现最佳的. 但是, 它仅仅是对于这个情形有用, 因为我们提供了密集分布的数据点. 另外众所

---

周知的是样条方法很难推广到多于三维的系统. 另外两种方法在高维情形有望比样条方法表现更好, 如果考虑到 RKHS 方法本身在高维情形所具有的优势.

**表 1** 不同插值方法在关键平衡位置点的构形和能量比较. 能量单位是 kcal/mol 相对于  $\text{N}+\text{H}_2$  平衡位置, 键长和键角的单位分别是波尔半径和度.

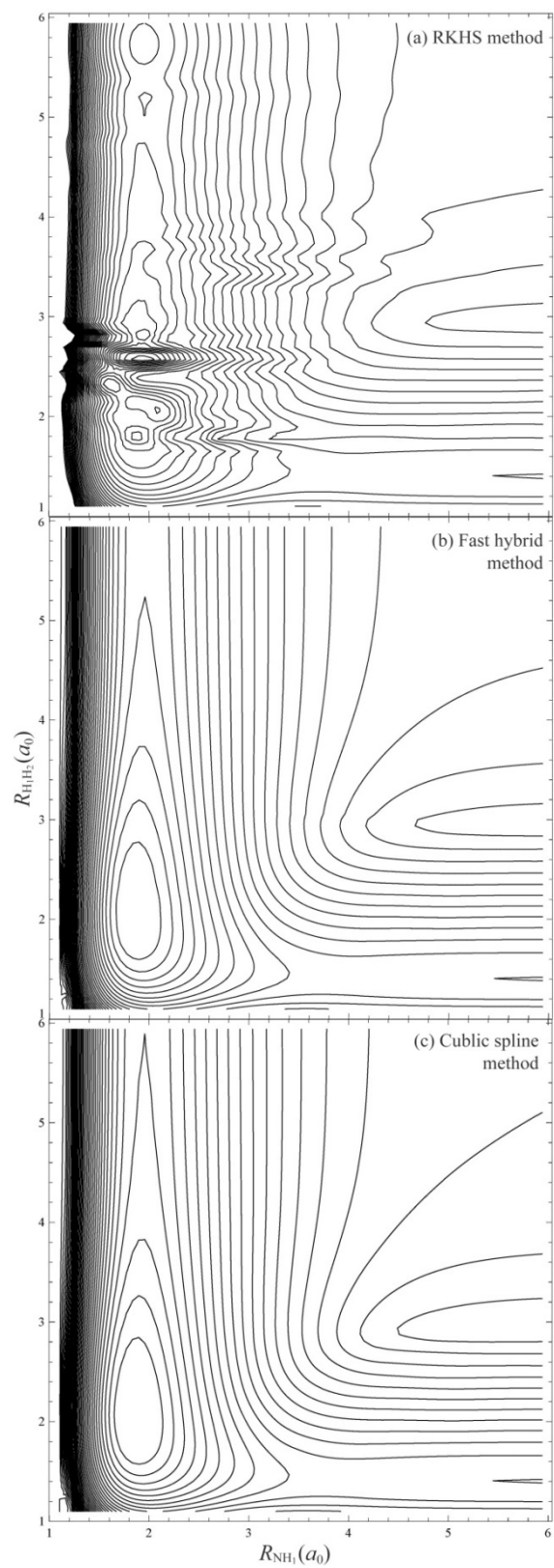
特征	属性	样条	混合
$\text{NH}_2$	$R_{\text{N-H}}$	1.923	1.927
	$R_{\text{H-H}}$	3.002	3.019
	$E$	-128.93	-128.94
$\text{N}+\text{H}_2$	$R_{\text{H-H}}$	1.402	1.402
	$E$	0.00	0.00
$\text{NH}+\text{H}$	$R_{\text{N-H}}$	1.958	1.958
	$E$	-29.30	-29.30
$\text{C}_{2v}$ 势垒	$R_{\text{H-H}}$	1.419	1.417
	$R_{\text{N-H}_2}$	3.944	3.941
	$E$	1.948	1.947

**表 2** 在不同能量上界下从头算点处混合方法的均方根 (RMS) 误差. 能量是相对于  $\text{N}+\text{H}_2$  平衡位置, RMS 误差单位是 kcal/mol.

能量上界	无穷大	7.0 eV	5.0 eV	3.0 eV
点数	25141	22256	21520	19822
RMS 误差	0.889	0.642	0.608	0.358

**表 3** 不同插值方法分别在启动阶段和求值阶段的耗时. 所有的计算都是基于  $\text{NH}_2$  系统的 25141 个从头算数据点. 求值阶段列列出的是求每个点的平均时间.

方法	启动阶段/秒	求值阶段/微秒
RKHS	5148.711	1.72
快速混合	18.972	2.81
样条 (慢)	0.121	87.03
样条 (快)	5.661	0.47



**图1** 从三种不同方法构建的  $\text{NH}_2$  势能面的等高线图, (a) 快速 RKHS 方法, (b) 快速混合方法, (c) 三维立方样条方法.  $\xi_{\text{NH}_1\text{H}_2}$  坐标被固定为值  $\pi/2$ . 横坐标和纵坐标分别表示 N-H 键长 (原子单位) 和 H-H 键长 (原子单位).

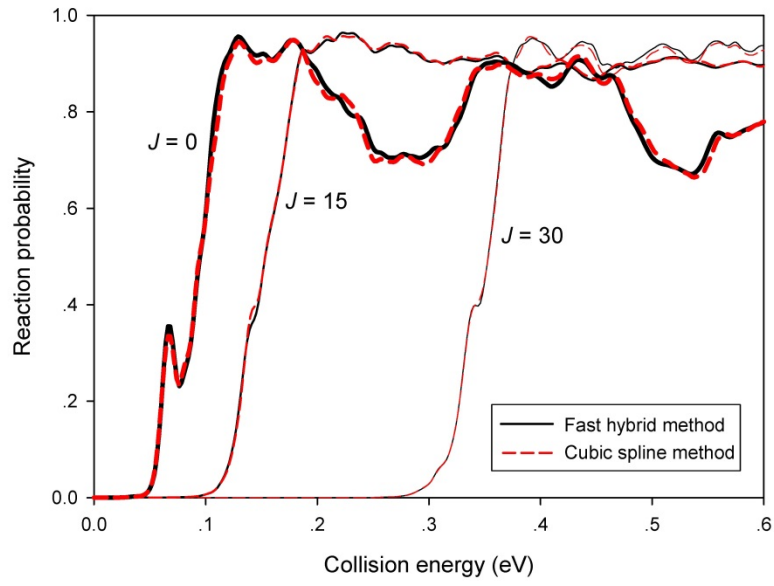
### 3.2 量子波包计算结果

为定量测试快速混合方法的准确性和可靠性, 我们使用由快速混合方法构建的势能面, 进行了精确三维量子动力学计算, 并得到了  $\text{N}(^2\text{D})+\text{H}_2\rightarrow\text{NH}+\text{H}$  反应在不同能量的反应概率和 ICS. 动力学结果和使用三维立方样条方法构建的势能面得到的结果 [35] 进行了比较.

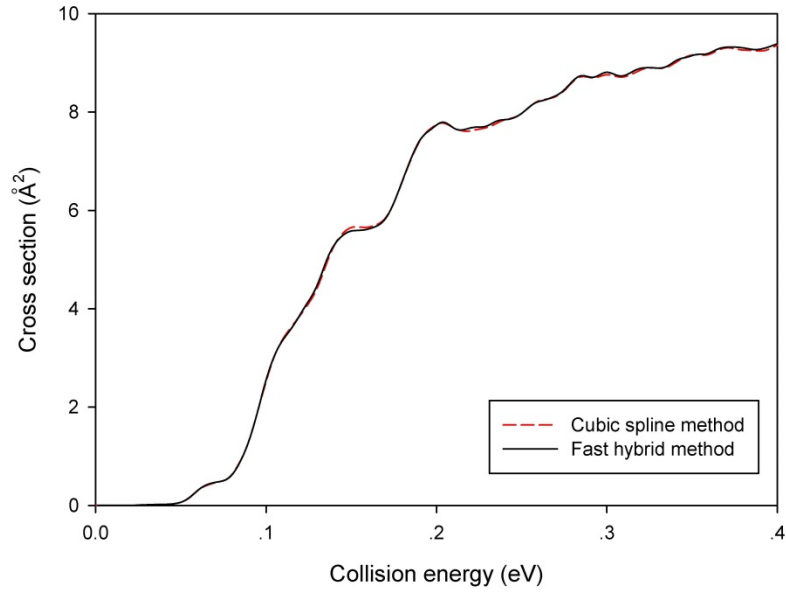
为了消除由于动力学计算造成的不确定性, 我们选择了足够大的数值参数来进行量子波包计算. 沿  $R$  ( $\in(0, 16)a_0$ ) 和  $r$  ( $\in(0.5, 8.0)a_0$ ) 坐标分别选择了 224 和 53 个等距格点. 利用了  $\text{H}_2$  分子的交换对称性将角度空间和转动基组的尺寸减小到原来的一半. 最终, 在角度坐标 ( $\gamma\in(0^\circ, 90^\circ)$ ) 选取了 25 个高斯-勒让德求积点, 并选择了由 25 个偶数转动角动量量子数组成的转动基组. 这种基组被证明是合适的. 在  $R = 13.0a_0$  和  $r = 5.0a_0$  处分别放置了一个高斯形的吸附函数, 衰减系数等于  $0.01a_0^{-2}$ . 反应流在  $r = 4.9a_0$  处计算. 对于所有的部分波, 进行了 15 000 步的传播. 通过对部分波进行求和得到了 ICS. 对所有的部分波都考虑了科里奥利耦合.

图 2 显示了固定初态的 ( $v_i = 0, j_i = 0$ ) 对于不同总角动量  $J$  的准确总反应速率, 分别针对两个不同的由快速混合方法构造的和三维立方样条方法构造的势能面. 两个势能面的结果显现出高度一致. 不仅仅是绝对值, 各处细节结构也都符合很好. 这些结果也和直接由从头算数据推算的特性一致, 比如, 由势垒导致的非零阈能, 深势阱导致的对能量的震荡相关性, 以及反应放热性对应的大的反应活性, 等等. 在 ICS 层次两种方法的一致性变得更加显著, 如图 3 所示.

很明显不幸的是我们没有办法的到一个给定离散从头算数据点集合的标准势能面. 但是, 如果两种不同拟合方法中的一种或两种是不准确的, 那么对于这两种势能面给出一致的动力学结果应当是极其不可能的. 这就意味着两个势能面动力学结果的高度吻合是混合方法准确性的极好证据.



**图 2** 初始态固定的 ( $v_i = 0, j_i = 0$ ) 对  $J=0, 15$ , 和  $30$  的总量子反应概率. 分别基于两个不同方法, 快速混合方法 (黑色实线) 以及三维立方样条方法 (红色虚线), 进行了计算.



**图 3** 初始态固定的 ( $v_i = 0, j_i = 0$ ) 准确量子 ICS 与能量的关系. 分别基于两个不同方法, 快速混合方法 (黑色实线) 以及三维立方样条方法 (红色虚线), 进行了计算.



---

## 第四章 结论

本文我们提出了一个新的混合方法用于从从头算数据点构建全局解析势能面. 这个方法结合了 RKES 方法和其他拟合方法, 比如立方样条插值, 并在划分的具有较小维度的构形空间应用不同的方法. 该方法可以很好地替代 RKHS 方法 [17] 因为纯 RKHS 方法在提供了大量从头算数据点时会变得不稳定. 而这种情形在多维系统中应该是很常见的. 混合方法提出了解决类似问题的一种思路, 即通过把分子构形空间分割为两个或多个子空间. 混合方法不仅仅继承了纯 RKHS 方法的大部分优势, 比如可以得到势能面的解析导数, 以及正确的渐近行为, 它还提供了许多附加的优势, 其中包括更快的启动时间和增强的准确性和稳定性.

作为一个特例, 我们对于  $\text{NH}_2$  系统从两万多个从头算数据点使用混合方法构建了一个三维势能面. 我们对于两个键长 N-H 和 H-H 构成的二维构形空间应用 RKHS 方法. 第三维沿着 N-H-H 键角则使用一维立方样条方法进行处理. 基于两个插值方法, RKHS 和样条方法的公式, 我们提出了一个混合方法的快速算法, 即通过预先计算和存储组合系数来使得计算势能值和它的导数可以更加快速.

为了检验混合方法的准确性, 我们基于构建的势能面进行了精确的三维量子动力学计算. 动力学计算显示混合方法对于构建高质量从头算势能面是非常可靠的, 它不仅适用于量子动力学, 还适用于经典轨线计算的研究.

尽管混合方法对于三体问题的计算是成功的, 对于将这一方法应用到更大的系统可能仍是具有挑战性的. 尽管看上去应用相同的思想处理是非常直接的, 但是会导致一些技术上的问题, 比如进行大量格点分布的从头算计算的可行性. 另外, 当对高维子空间应用 RKHS 方法, 线性系统可能仍会是病态的. 所以我们需要谨慎选择从头算数据点的位置. 比如, 以四原子系统为例, 这是一个 6 维系统, 我们的建议是在对动力学计算较为关键的区域选择相对密集的格点, 这些区域可以是与键断开过程相关的区域. 我们在这些区域的子空间进行立方样条插值. 对于剩余的维度我们可以用 RKHS 插值, 并且在这些点选取较少的从头算数

---

据点.

## 参考文献

- [1] J.N. Murrell, S. Carter, S.C. Farantos, P. Huxley, A.J.C. Varandas, *Molecular Potential Energy Functions*, Wiley, Chichester, 1984.
- [2] G.C. Schatz, *Rev. Mod. Phys.* 61 (1989) 669.
- [3] B.J. Braams, J.M. Bowman, *Int. Rev. Phys. Chem.* 28 (2009) 577.
- [4] T. Hollebeek, T.-S. Ho, H. Rabitz, *Annu. Rev. Phys. Chem.* 50 (1999) 537.
- [5] N. Sathyamurthy, L.M. Raff, *J. Chem. Phys.* 63 (1975) 464.
- [6] J.M. Bowman, J.S. Bittman, L.B. Harding, *J. Chem. Phys.* 85 (1986) 911.
- [7] C. Xu, D. Xie, D.H. Zhang, S.Y. Lin, H. Guo, *J. Chem. Phys.* 122 (2005) 244305.
- [8] R. Dawes, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* 128 (2008) 084107.
- [9] Y. Guo, A. Kawano, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* 121 (2004) 5091.
- [10] G.G. Maisuradze, D.L. Thompson, *J. Phys. Chem. A* 107 (2003) 7118.
- [11] J. Behler, *Phys. Chem. Chem. Phys.* 13 (2011) 17930.
- [12] D.F.R. Brown, M.N. Gibbs, D.C. Clary, *J. Chem. Phys.* 105 (1996) 7597.
- [13] S. Manzhos, T. Carrington, *J. Chem. Phys.* 125 (2006) 084109.
- [14] J. Chen, X. Xu, X. Xu, D.H. Zhang, *J. Chem. Phys.* 138 (2013) 154301.
- [15] J. Li, B. Jiang, H. Guo, *J. Chem. Phys.* 139 (2013) 204103.
- [16] K.C. Thompson, M.J.T. Jordan, M.A. Collins, *J. Chem. Phys.* 108 (1998) 8302.
- [17] T.S. Ho, H. Rabitz, *J. Chem. Phys.* 104 (1996) 2584.
- [18] X. Zhang, B.J. Braams, J.M. Bowman, *J. Chem. Phys.* 124 (2006) 021104.

- 
- [19] T.S. Ho, T. Hollebeek, H. Rabitz, L.B. Harding, G.C. Schatz, J. Chem. Phys. 105 (1996) 10472.
- [20] L.A. Pederson, G.C. Schatz, T.-S. Ho, T. Hollebeek, H. Rabitz, L.B. Harding, G. Lendvay, J. Chem. Phys. 110 (1999) 9091.
- [21] T. Hollebeek, T.-S. Ho, H. Rabitz, J. Chem. Phys. 106 (1997) 7223.
- [22] S. Arora, B. Barak, Computational complexity: A modern approach, Cambridge University Press, 2009.
- [23] M.T. Heath, Scientific Computing: An Introductory Survey, McGraw-Hill, 1997.
- [24] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, NUMERICAL RECIPES IN FORTRAN 77, Cambridge University Press, Cambridge, 1992.
- [25] M. Karplus, R.N. Porter, R.D. Sharma, J. Chem. Phys. 43 (1965) 3259.
- [26] Y. Huang, D.J. Kouri, D.K. Hoffman, Chem. Phys. Lett. 225 (1994) 37.
- [27] V.A. Mandelshtam, H.S. Taylor, J. Chem. Phys. 102 (1995) 7390.
- [28] V.A. Mandelshtam, H.S. Taylor, J. Chem. Phys. 103 (1995) 2903.
- [29] S.K. Gray, G.G. Balint-Kurti, J. Chem. Phys. 108 (1998) 950.
- [30] H. Guo, Int. Rev. Phys. Chem. 31 (2012) 1.
- [31] R. Chen, H. Guo, J. Chem. Phys. 105 (1996) 3569.
- [32] S.Y. Lin, H. Guo, J. Phys. Chem. A 108 (2004) 2141.
- [33] J.C. Light, T. Carrington Jr., Adv. Chem. Phys. 114 (2000) 263.
- [34] S.Y. Lin, H. Guo, J. Chem. Phys. 119 (2003) 11602.
- [35] S. Zhou, D. Xie, S.Y. Lin, H. Guo, J. Chem. Phys. 128 (2008) 224316.
- [36] J.N. Murrell, S. Carter, J. Phys. Chem. 88 (1984) 4887.