arXiv:2301.11653v2 [eess.SY] 3 Jan 2024

通过线性变换对动力学模型进行精确分层约化

Alexander Demin[a,*]、Elizaveta Demitraki[a]、Gleb Pogudin[b]

[a]俄罗斯莫斯科国立研究型大学、国立高等经济大学[b]法国帕莱索巴黎理工学院巴黎综合理工学院*LIX*，法国国家科学研究中心

摘要

由常微分方程（ODE）描述的动力学模型是科学与工程领域的基础工具。精确约简旨在构建一个低维模型，使其中的每个宏变量都能直接关联到原始变量，因此它是实现模型形式分析和机理理解的自然步骤。我们提出一种算法，该算法以多项式常微分方程模型为输入，计算模型可能存在的最长精确线性约化链，其中每次约化都细化前一次约化，从而使用户能够控制约化所保留的细节层次。这显著推广了现有方法：这些方法仅在特定于各自方法的约束下计算维数最低的约化。该算法将寻找精确线性约化的问题转化为关于有限维代数表示的问题。我们提供了该算法的实现，在一组基准测试上展示其性能，并通过案例研究说明其适用性。我们的实现可从以下地址免费下载：
https://github.com/x3042/ExactODEReduction.jl。

关键词： 常微分方程、精确约简、集总、降维、矩阵代数*2000*年数学主题分类： 34C20,34-04, 16G10

1. 引言

常微分方程（ODEs）为描述实时演化的系统提供了一种强大且富有表达力的语言，因此广泛应用于科学和工程领域。这促使人们开发形式方法，以分析使用常微分方程定义的模型结构。过去十年中，从这一角度得到积极研究的一个重要问题是模型约化 [1, 2, 3, 4, 5]。

*通讯作者电子邮箱： alexander.demin.eternal@gmail.com (AlexanderDemin), egdemitraki@edu.hse.ru (Elizaveta Demitraki), gleb.pogudin@polytechnique.edu (Gleb Pogudin)

提交至爱思唯尔的预印本 *2024*年*1*月*5*日

一般而言，模型约化是指一类技术，其目标是用一个低维模型替代所关注的模型，并借此分析原始模型的动力学。传统上，人们采用了诸如平衡截断等近似方法[6] 。精确模型约化则是一种互补方法，它在不引入近似误差的情况下降低模型的维数。由于具有精确性，此类约化通常与系统结构相关，因此在进行形式分析或获得机制性见解的背景下尤其值得关注。精确模型约化的经典例子包括利用守恒律进行的约化，或基于对称性的约化 [7, 8, 9]。原则上，精确变换除可降低模型维度外，也可用于降低模型复杂度的不同度量。例如，精确变换可用于将模型转换为线性模型 [10]。此类变换不在本文的讨论范围之内。

用于寻找精确降维的算法可以直接应用于常微分方程组（例如本文提出的算法），也可以应用于模型的领域专用描述（随后可将其转换为常微分方程组）。基于规则的建模 [11, 12] 是此类简洁领域专用描述的一个典型例子，针对这类描述，人们已设计出强大的约简算法 [1, 8, 13] 并将其实现 [14]。与本文考察的基于常微分方程的算法相比，依赖基于规则的表示的算法能够处理非常大型的模型，但仅适用于一类常微分方程组，且只搜索一类约简（通过所谓的片段 [1], 另请参见注释1）。

本文将重点讨论常微分方程模型的一类重要精确约简：精确线性聚合。这类约简旨在为一组宏变量寻找一个封闭的微分方程组，使得每个宏变量都是原始变量的线性组合。将宏变量设为原始变量之和的情形已受到广泛关注，参见例如 [1, 15, 2, 4]。尤其是，人们已开发出 ERODE 软件 [3] ，能够高效地找出变量划分为宏变量的最优方案。CLUE 软件包 [5, 16] 则是在放宽宏变量限制方面迈出的一步。与早期方法相比，CLUE 找出的宏变量可以包含任意系数（而不再仅限于零和一），并且允许同一个变量同时出现在多个宏变量中。因此，CLUE 中约化模型的维数可能显著降低 [5, 表 1]。然而，该算法的输入不仅包括模型，还包括需要保留的状态变量的线性函数（即可观测量）。这类可观测量可能存在，也可能不存在；能否正确猜出它们，对于找到低维约化至关重要。本文旨在兼取两种方法之长：输入只需提供模型（如 ERODE），同时允许宏变量为原始变量的任意线性组合（如 CLUE）。

本文的主要成果是一种算法，仅给定一个具有有理系数的多项式常微分方程模型，就能找到任意精确线性约化。请注意，寻找维数尽可能低的此类任意线性集总

未必最有意义，因为任何线性第一积分都会产生一个维数为一、具有恒定动力学的约化。相反，我们的算法构建了一个约化层级，更准确地说，它 找出可能最长的聚合链，其中每个约化都是下一个约化的细化（详情见第 2 节）。这样，用户就可以沿着层级选择约化所要保留的细节层级，并且可能找到 ERODE 和 CLUE 很可能会遗漏的约化（例如，参见示例 3 和第 5.2 节）。这种通用性也有代价：我们的软件通常比 CLUE 和 ERODE 慢。

我们的算法基于两者的结合：一是线性集总问题与求解一组矩阵 [17, 5] 的公共不变子空间问题之间的联系，二是有限维代数的结构理论。我们采用现有算法在有限 [18] 域和代数闭 [19, 20] 域上的通用框架，并通过以下方法实现所需的效率：

- 一种用于求代数的一组基的稀疏性感知算法（第 3.2节）；
- 利用输入的结构，尽可能主要使用有理数进行计算，并尽可能推迟转入代数数域；
- 分别使用稀疏线性代数和模计算来处理大型矩阵和表达式膨胀问题。

我们实现了该算法，实现代码已公开发布于 https://github.com/x3042/ExactODEReduction.jl。我们使用来自 BioModels 数据库的一组基准测试来评估其性能 [21], 该数据库收录了大量生命科学模型；此外，我们还展示了该算法在两个案例研究中得到的约化结果。

本文其余部分安排如下。在第2节中，我们给出精确线性约化的准确定义，并明确表述本文要解决的算法问题。第3 节详细介绍算法及其论证。我们将在第4节介绍算法的实现并报告其性能。第5 节介绍案例研究，说明软件所生成的约化。

2. 问题陈述

本文中，矩阵 M 的转置记为 M^T。对于由不定元组成的向量 $\mathbf{x} = (x_1, \ldots, x_n)$，我们用 $\mathbb{C}[\mathbf{x}]$（分别用 $\mathbb{R}[\mathbf{x}]$、$\mathbb{Q}[\mathbf{x}]$）表示 $\mathbf{x}$ 中系数为复数（分别为实数、有理数）的多项式集合。

定义 1（集总）。考虑一个常微分方程组

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \tag{1}$$

对具有多项式右端的变量 $\mathbf{x} = (x_1, \ldots, x_n)$，即 $\mathbf{f} = (f_1, \ldots, f_n)$ 和 $f_1, \ldots, f_n \in \mathbb{C}[\mathbf{x}]$。我们称线性变换 $\mathbf{y} = \mathbf{x}L$

满足 $\mathbf{y}=(y_1,\dots,y_m)$， $L\in\mathbb{C}^{n\times m}$，且 秩 $L=m$ 是 （*1*）的集总(1)，当且仅当存在 $\mathbf{g}=(g_1,\dots,g_m)$ 满足 $g_1,\dots,g_m\in\mathbb{C}[\mathbf{y}]$ 使得

$$\mathbf{y}'=\mathbf{g}(\mathbf{y})$$

对于（1）的每个解$\mathbf{x}$， 1中的数值 m 称为集总的维数，而$\mathbf{y}$ 的各个分量称为宏变量。

注意，集总仅由矩阵 L 唯一确定，相应的$\mathbf{g}$ 可以计算得到，详见备注3 。

本节中，我们将以以下示例为例 [22, 示例 1]。该常微分方程模型将根据质量作用动力学定律 [23, 第 7] 章，从一个包含化学物种 X以及另外四种物种 A_{UU}、 A_{UX}、 A_{XU}和 A_{XX}的化学反应网络推导得出。这些物种表示一个分子 A， 它有两个相同的结合位点，每个位点可以是未结合状态（记为 U），也可以与 X 结合（记为 X）。这些物种满足以下反应

$$X+A_{U*}\underset{k_2}{\overset{k_1}{\rightleftarrows}}A_{X*}\qquad X+A_{*U}\underset{k_2}{\overset{k_1}{\rightleftarrows}}A_{*X}\tag{2}$$

其中 $*\in\{X,U\}$，结合和解离的反应速率常数分别为 k_1 和 k_2。若将任意物种 S 的浓度$[S]$记为，根据质量作用动力学定律得到的相应常微分方程组 [23, 第 7]章为:

$$\begin{cases}[X]'=k_2([A_{XU}]+[A_{UX}]+2[A_{XX}])-k_1[X]([A_{XU}]+[A_{UX}]+2[A_{UU}]),\\ [A_{UU}]'=k_2([A_{XU}]+[A_{UX}])-2k_1[X][A_{UU}],\\ [A_{UX}]'=k_2([A_{XX}]-[A_{UX}])+k_1[X]([A_{UU}]-[A_{UX}]),\\ [A_{XU}]'=k_2([A_{XX}]-[A_{XU}])+k_1[X]([A_{UU}]-[A_{XU}]),\\ [A_{XX}]'=k_1[X]([A_{XU}]+[A_{UX}])-2k_2[A_{XX}].\end{cases}\tag{3}$$

例 1（守恒律作为集总）。我们证明，矩阵$L=(0\ 1\ 1\ 1)^T$ 给出了（3）的一个集总。我们有

$y=\begin{pmatrix}[X] & [A_{UU}] & [A_{UX}] & [A_{XU}] & [A_{XX}]\end{pmatrix}\cdot L=[A_{UU}]+[A_{UX}]+[A_{XU}]+[A_{XX}].$

利用（3）可以验证 $y'=0$，因此可以取 $g(y)=0$。确实， y 是 A 的总浓度，必须保持恒定。此外，任意线性守恒律都会产生一个维数为一的集总。

例2（信息量更丰富的集总）。同一系统（3）的另一种集总由矩阵给出

$$L=\begin{pmatrix}1&0&0\\0&2&0\\0&1&1\\0&1&1\\0&0&2\end{pmatrix}\implies\begin{cases}y_1=[X],\\ y_2=2[A_{UU}]+[A_{UX}]+[A_{XU}],\\ y_3=[A_{UX}]+[A_{XU}]+2[A_{XX}].\end{cases}$$

宏变量将满足一个自包含的系统

$$y_1' = k_2 y_3 - k_1 y_1 y_2, \quad y_2' = k_2 y_3 - k_1 y_1 y_2, \quad y_3' = -k_2 y_3 + k_1 y_1 y_2.$$

这一约化的依据是，y_2 和 y_3 分别是未结合位点和已结合位点的浓度。

以上示例表明，同一个系统可以有多个集总（事实上，（3）有更多），因此自然会提出一个问题：如何找到有用的集总。现有的先进软件工具CLUE [5] 和ERODE [3] 通过寻找满足特定约束且维数最小的集总来解决这一问题：

- 保留某些感兴趣的量，不将其集总（对于 CLUE，[5]）；
- 或由状态变量的一个划分得到（对于 ERODE，[3]）。

这两种约束可能过于严格：并非所有有趣的集总都来自状态变量的划分（例如，例1 和2；另见 [5,表 1]），而且预先猜出有意义的保留量可能很困难。

例3（对CLUE 和ERODE而言较难的示例，另见第5.2节）。考虑另一个化学反应网络 [24, 方程（19.20）]（最初由 Daniel Knight 提出）：

$$\begin{array}{ccccc} E+S & \underset{k_2}{\overset{k_1}{\rightleftharpoons}} & ES & \underset{k_4}{\overset{k_3}{\rightleftharpoons}} & E+P \\ & & \downarrow k_5 & & \\ & & E^*+S & & \end{array} \qquad E \underset{k_6}{\overset{k_5}{\rightleftharpoons}} E^*$$

其中，我们假设 $ES \to E^* + S$ 和 $E \to E^*$ 的速率常数相等。

与（2）的情况一样，我们利用质量作用动力学定律将反应转化为常微分方程组：

$$\begin{cases} [E]' = (k_2 + k_3)[ES] + k_6[E^*] - k_1[E][S] - k_4[E][P] - [E], \\ [S]' = (k_2 + k_5)[ES] - k_1[E][S], \\ [P]' = k_3[ES] - k_4[E][P], \\ [ES]' = k_1[E][S] + k_4[E][P] - (k_2 + k_3 + k_5)[ES], \\ [E^*]' = k_5[E] + k_5[ES] - k_6[E^*] \end{cases}$$

一种有意义的线性约化是 $y = [E] + [ES] - \frac{k_6}{k_5}[E^*]$，其方程为 $y' = -(k_5 + k_6)y$。宏变量 y 可以理解为 E（通常是酶）的量（包括游离形式 E 以及作为复合物 ES一部分的形式）与 E^*（通常是失活酶）的量之间的势。两种量之间存在双向流动，因此任意 k_5 单位的 $[E] + [ES]$都会与 k_6 单位的 $[E^*]$处于平衡状态，因此差值 y 的动力学仅取决于其自身。

这种约化并非源自对物种的划分，因此无法通过 ERODE找到。此外，若要使用 CLUE 找到它，就需要预先知道这个宏变量。由于当前版本的软件不允许宏变量的系数中出现符号参数（参见备注 6），我们通过取 k_5和 k_6 的数值来找到了这一约化。

一种替代方法是找出所有集总，并让用户选择要使用哪些。问题在于，集总的数量很容易是无限的。例如，类似于例1，可以证明矩阵

$$L = \begin{pmatrix} \alpha & 1 & 1+\alpha & 1+\alpha & 1+2\alpha \end{pmatrix}^T$$

对每个数 α都给出（3）的一个集总。此外，正如我们稍后将要说明的，集总与某些矩阵的不变子空间构成双射，这些矩阵来自$\mathbf{f}(\mathbf{x})$ 的雅可比矩阵；而且（至少对于任意矩阵而言），不变子空间可以构成任意代数簇 [25]。

本文采用的方法是寻找一系列彼此逐步细化的约化，并保证该序列的长度尽可能长。

定义 2（聚合链）。对于形式为 $\mathbf{x}'$ =$\mathbf{f(x)}$ 的常微分方程组，一系列线性变换

$$\mathbf{y}_1 = \mathbf{x}L_1,\ \mathbf{y}_2 = \mathbf{x}L_2,\ \ldots,\ \mathbf{y}_\ell = \mathbf{x}L_\ell,$$

其中 $L_1 \in \mathbb{C}^{n\times m_1}, \ldots, L_\ell \in \mathbb{C}^{n\times m_\ell}$, 若满足以下条件，则称为聚合链

1. $0 < m_1 < \ldots < m_\ell < n$;
2. $\mathbf{y}_i$ =$\mathbf{x}L_i$ 是（1）对每个 $1 \leqslant i \leqslant \ell$的集总;
3. 对每个 $1 < i \leqslant \ell$，都存在一个矩阵 A_i 使得 $L_{i-1} = L_i A_i$。

后者意味着由 L_1、.....、L_ℓ 给出的约化彼此细化。这样的链（L_1，.....，L_ℓ）若不作为子序列包含在任何更长的链中，则称为极大链。

我们将证明（见推论2），所有极大链的长度都相同，因此它们也是可能的最长链。给定一条集总的极大链，用户可以根据对约化模型规模和保留信息量之间权衡的需求，沿链向左或向右“放大/缩小”。因此，我们现在可以正式陈述本文研究的主要问题。

主要问题 1。

给定一个系统 $\mathbf{x}' = \mathbf{f}(\mathbf{x})$，其中 $\mathbf{f}$ 是由 $\mathbb{Q}$ 中的多项式构成的向量;

计算该系统的最大集总链。

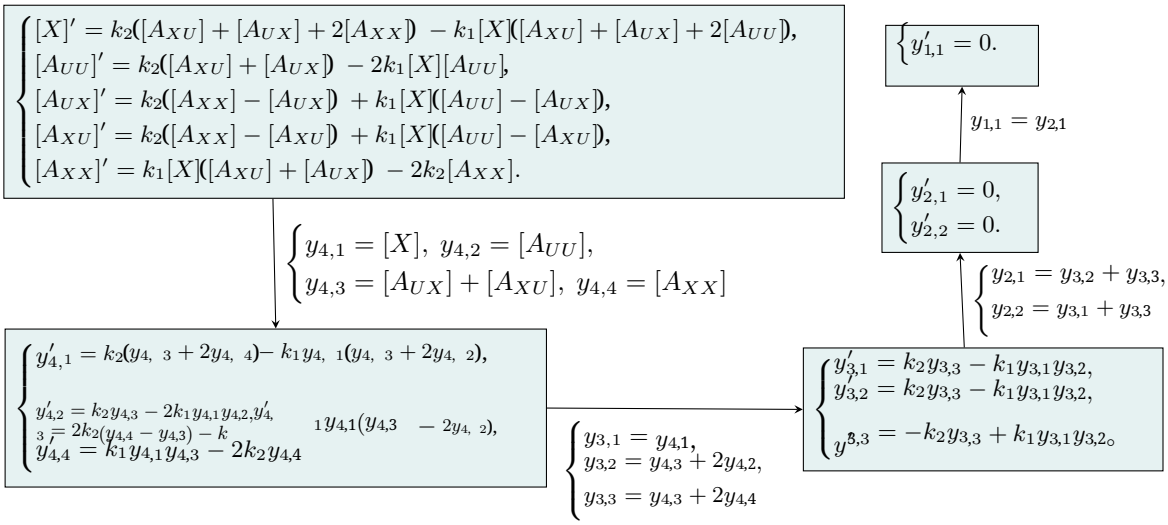


图1：对于（3）的最大集总链及相应的约化

例 4（(3)的最大集总链）。图1 展示了我们示例系统（3）的一条集总链及其相应的约化。块中包含约化系统，箭头标出了连续约化之间的变换（矩阵 A_i，其形式见定义2）。该约化链包括前面的例1 和例2，分别对应于$\mathbf{y}_1$ 和$\mathbf{y}_3$，。

在此例中，原始维数为 $n = 5$，约化的维数分别为 $m_1 = 1$、$m_2 = 2$、$m_3 = 3$、$m_4 = 4$，因此这条链显然是极大的。

注1（与基于规则的表示的对称性的联系）

（2）中的模型自然地符合基于规则的建模框架 [11,12]。该模型由规定物种动力学的规则构成，而这些规则本质上是由不同的 $* \in \{X, U\}$参数化的化学反应。请注意，这些规则关于分子 A的两个结合位点是对称的。得益于这种对称性，可以直接在规则层面进行约化，方法是按照结合位点总数对物种分组，如下所示：

$$y_1 = [X], \quad y_2 = [A_{UU}], \quad y_3 = [A_{UX}] + [A_{XU}], \quad y_4 = [A_{XX}].$$

请注意，这是图 1所示约化链中的第一个约化。存在能够利用此类基于规则的对称性的算法 [13],，而这一特定约化可以通过 KaDE 软件工具 [14][1]找到。下一个约化

[1]我们感谢一位匿名审稿人提醒我们注意这一事实。

然而，链中的下一个约化并非由 KaDE发现的，尽管它仍可从底层反应网络的结构角度加以解释 [22,例 1 和第 3 节。1]我们在此看到了协同增效的机会：我们的结构无关工具或许能发现新类型的约化，其中一些类型随后可以纳入可扩展的基于规则的方法。

注释 2（与箭图等变常微分方程的联系）。可以在箭图等变动力系统的框架下考察聚合链 [26]，其中相应的箭图是一条链，箭上的映射由 A_i 定义。基于这一视角，上文提出的主要问题可以自然地推广为：寻找一个极大（在某种意义上）的箭图 $\mathcal{Q}$，使得原常微分方程可以表示为 $\mathcal{Q}$-等变形式。

3. 算法

为寻找最大集总链，我们首先运用 [5] 中发展的理论，将问题约化为一组矩阵的公共不变子空间问题（第 3.1 节），然后利用有限维代数的结构理论解决这一新问题（第 3.3 至 3.5 节）。整体算法总结于第 3.6 节。

3.1. 约化为寻找公共不变子空间

设 $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ 是一个变量为 $\mathbf{x} = (x_1, \ldots, x_n)$、且 $\mathbf{f}$ 为行向量的常微分方程组，其中各分量为多项式 $f_1, \ldots, f_n \in \mathbb{C}[\mathbf{x}]$。设 $J(\mathbf{x})$ 为 $\mathbf{f}$ 关于 $\mathbf{x}$ 的雅可比矩阵。我们用 $m_1(\mathbf{x}) \ldots, m_N(\mathbf{x})$ 表示出现在 $J(\mathbf{x})$ 中的 $\mathbf{x}$ 的单项式。于是 $J(\mathbf{x})$ 可唯一写成

$$J(\mathbf{x}) = \begin{pmatrix} \nabla f_1 & \ldots & \nabla f_n \end{pmatrix} = \sum_{i=1}^{N} J_i \cdot m_i(\mathbf{x}), \quad \text{where } \nabla g := \begin{pmatrix} \frac{\partial g}{\partial x_1} & \ldots & \frac{\partial g}{\partial x_n} \end{pmatrix}^T \tag{4}$$

以及 J_1、……、J_N 为常矩阵。

例 5。考虑系统

$$x_1' = x_1 - 2x_2^2, \quad x_2' = -x_2 + x_2^2.$$

在此情况下，分解（4）为

$$J(x_1, x_2) = \begin{pmatrix} 1 & 0 \\ -4x_2 & -1 + 2x_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -4 & 2 \end{pmatrix} x_2.$$

引理 1。使用上述记号，线性变换 $\mathbf{y} = \mathbf{x}L$（其中 $L \in \mathbb{C}^{n \times m}$）是 $\mathbf{x}' = \mathbf{f}(\mathbf{x})$ 的集总，当且仅当 L 的列空间关于 J_1、……、J_N 不变。

证明。 对于 $L \in \mathbb{R}^{n\times m}$ 的情形，该陈述由 [5, 引理 S.I.1 和 S.II1] 得出。 [5, 引理 S.I.1] 的证明在将 $\mathbb{R}$ 替换为$\mathbb{C}$后仍然成立，而 [5, 引理 S.II.1] 的证明在 $\mathbb{C}$ 的情形下，只需将实内积替换为复内积即可成立。 □

注 3. 一个自然的问题是：给定满足引理 1 条件的 L，如何求出定义 1 中的 g? 一种方法如下：令 $\mathrm{y} := \mathrm{x}L$，并选取 x 的一个子集 $\tilde{}\mathrm{x}$，将 y 补充为线性张成空间的一组基。然后，导数 y'（等于 $\mathrm{f}L$）可通过线性坐标变换表示为 y 和 $\tilde{}\mathrm{x}$ 的函数。引理 1 保证这些多项式实际上不会依赖于 $\tilde{}\mathrm{x}$，因此它们恰好就是 g。CLUE [5], 中已使用了这一构造的优化版本，我们的实现中也采用了该版本。

推论 1. 一系列线性变换 $\mathrm{y}_1 = \mathrm{x}L_1, \ldots, \mathrm{y}_\ell = \mathrm{x}L_\ell$ 是一个聚合链，当且仅当 $L_1, \ldots, L_\ell$ 的列空间 $V_1, V_2, \ldots, V_\ell$ 满足

- V_i 对于每个 $1 \leqslant i \leqslant \ell$，均关于 J_1、……、J_N 保持不变；
- $\{0\} \subsetneq V_1 \subsetneq \ldots \subsetneq V_\ell \subsetneq \mathbb{C}^n$.

此外，当且仅当聚合链 V_1、……、V_ℓ 不是满足上述两个性质的子空间链的子序列时，该聚合链才是极大的。

为了寻找这样的不变子空间链，我们将运用有限维矩阵代数理论。

定义 3（矩阵代数）。设 k 是一个域（例如 $k = \mathbb{Q}$、$\mathbb{R}$、$\mathbb{C}$）。矩阵的子空间 $\mathcal{A} \subseteq k^{n\times n}$ 称为代数，如果它在乘法运算下封闭且包含单位矩阵。

对于有限集合 A_1、……、$A_m \in k^{n\times n}$，我们用 $\langle A_1$、……、$A_m\rangle$表示包含 A_1、……、A_m 的最小代数。该代数等于这些矩阵所有可能乘积的张成空间。

对于常微分方程组 $\mathrm{x}' = \mathrm{f(x)}$，其中 $\mathrm{x} = (x_1, \ldots, x_n)$ 且 $f_1, f_2, \ldots, f_n \in \mathbb{C}[\mathrm{x}]$,，我们考虑 f(x) 的雅可比矩阵按 x 的多项式形式表示时的系数 $J_1, \ldots, J_N$，如式 (4) 所示。我们将代数 $\langle I_n, J_1, \ldots, J_N\rangle$（其中 I_n 是单位 $n \times n$-矩阵）称为系统 $\mathrm{x}' = \mathrm{f(x)}$ 的雅可比代数。

例 6。

- 设 T_n 是 $k^{n\times n}$ 中所有上三角矩阵构成的集合。由于两个上三角矩阵的乘积仍是上三角矩阵，T_n是一个代数。

- 考虑例 5中的系统。其雅可比代数为

$$\left\langle \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ -4 & 2 \end{pmatrix} \right\rangle = \{M^T \mid M \in T_2\}.$$

由于子空间 $V \subset \mathbb{C}^n$ 相对于 $J_1, \ldots, J_N$ 是不变的，当且仅当它相对于 $\langle I_n, J_1, \ldots, J_N \rangle$ 是不变的，也就是说，它相对于代数中的任意元素都是不变的，因此我们将进一步研究如何求出该雅可比代数的不变子空间。一个直接的好处是，我们可以利用乔丹-赫尔德定理 [27, 定理 1.5。1]来明确我们对聚合极大链（定义 2）的理解：定义仅要求极大链不能再细化，但一般而言，这并不排除存在更长的链。以下由 [27, 定理 1.5。1] 直接推出的结论保证了极大链确实具有可能的最大长度。

推论 2. 对于给定的常微分方程组 $\mathrm{x}' = \mathrm{f(x)}$，所有聚合的极大链长度都相同。

3.2. 生成代数

为了对雅可比代数 $\langle I_n, J_1, \ldots, J_N \rangle$(定义 3) 进行显式计算，我们将计算其基。算法 1 给出了我们方法的一个简化版本，其本质上是 [5, 算法 2] 应用于矩阵而非向量。与 [5], 类似，我们采用模计算（参见 [5, 算法 3]），以避免中间过程中的表达式膨胀，并使用稀疏线性代数。

在对 [5], 所述方法进行这一直接改编的基础上，我们进一步利用输入和输出的稀疏性，显著提升了性能。在文献中的模型里，模型的非线性部分通常是稀疏的，因此，大多数 J_1，...，J_N 都极其稀疏；此外，雅可比代数的基通常也可以选得非常稀疏。因此，C 中**（步骤 2）** (b)i 得到的许多矩阵也会是稀疏的。然而，在中间步骤计算出的一些矩阵仍可能相当稠密，从而拖慢整个算法。为解决这一问题，我们暂时推迟对相对稠密的矩阵**（步骤 2）** (b)ii 的处理，C待外层循环结束、表明 P 为空后，再将每个暂缓处理的矩阵加入 P 并重新开始迭代。这样，我们便能确保在代数中生成足够多的稀疏矩阵，使得现在对稠密矩阵进行约化时，结果会更加稀疏。得益于这一优化，算法1 不再像直接采用 [5] 中的方法时那样成为计算瓶颈。

算法 1 寻找矩阵代数的一组基（基础版本）

输入一组方阵 $A_1, \ldots, A_\ell \in k^{n\times n}$;
输出 最小线性子空间 $\mathcal{A} \subseteq k^{n\times n}$ 的一组基 S，其中 $\mathcal{A} \subseteq k^{n\times n}$ 包含 A_1、...、...、A_ℓ的所有可能乘积;

（步骤 1）令 S为 A_1、.....、A_ℓ的线性张成空间的任意一组基，并令 $P\colon = S$。

（步骤 2）当 $P \neq \varnothing$ 时执行

（a）取 P中的一个元素 B，并将其从 P中移除。（b）对于 $\{A_1, \ldots, A_\ell\}$中的每个 A，执行 i. 计算 $C\colon = AB$，并通过高斯消元法相对于 S 对 C 进行约化。ii. 若 $C \neq 0$，令 $S\colon = S \cup \{C\}$，并令 $P\colon = P \cup \{C\}$。

（步骤 3）返回 S。

3.3. 不变子空间搜索：代数预备知识

本节中，我们固定一个特征为零的基域 k。我们主要关注的情形是有理数 $\mathbb{Q}$、代数数 $\overline{\mathbb{Q}}$ 和复数 $\mathbb{C}$。

定义 4（代数的根基）。设 $\mathcal{A} \subseteq k^{n\times n}$ 是一个代数。

- 若对每个 $A \in \mathcal{A}$ 和 $B \in \mathcal{I}$，都有 $AB, BA \in \mathcal{I}$（相应地，$AB \in \mathcal{I}$），则称子空间 $\mathcal{I} \subseteq \mathcal{A}$ 为理想（相应地，左理想）。
- 若存在 N，使得 $\mathcal{I}$ 中任意 N 个元素的乘积为零，则称理想（相应地，左理想）$\mathcal{I} \subseteq \mathcal{A}$ 为幂零。
- 所有满足以下条件的元素的集合：$A \in \mathcal{A}$ 左理想 $\mathcal{A}\cdot A$ 是幂零的，称为根基 $\mathcal{A}$。根据定理 3.1.6、3.1。$\mathcal{A}$，它是[27, 的一个幂零理想10]。

例 7. 设 T_n 是 $k^{n\times n}$ 中所有上三角矩阵构成的集合。考虑严格上三角矩阵的一个子集 $U_n \subset T_n$。容易验证，U_n 是一个理想，且 U_n 中任意 n 个元素的乘积均为零。由于对于每个 $A \in U_n$，都有 $T_n \cdot A \subseteq U_n$，因此我们推知 U_n 是 T_n 的根基。

Dixon 定理 [28, 定理 11] 表明，代数 $\mathcal{A} \subseteq k^{n\times n}$ 的根基可以通过求一个阶数为 $\dim \mathcal{A} \leqslant n^2$的方阵的核来计算。下面的引理说明了根基这一概念与我们的问题有何关联。

引理 2. 设 $\mathcal{A} \subseteq k^{n\times n}$ 为一个代数，$\mathcal{R} \subset \mathcal{A}$ 为其根基。若$\mathcal{R} \neq \{0\}$，则交 $\bigcap_{R\in\mathcal{R}}$ 核 R 非零，且关于 $\mathcal{A}$是不变的。

证明。由于 $\mathcal{R}$ 是幂零理想，存在最小整数 N，使得 $\mathcal{R}$ 中任意 N 个元素的乘积都为零。于是，存在 $0 \neq M \in k^{n\times n}$，它是 $\mathcal{R}$ 中 $N-1$ 个元素的乘积。因此，对于每个 $R \in \mathcal{R}$，都有 $RM = 0$，所以 $V := \bigcap_{R\in\mathcal{R}} \operatorname{Ker} R \supseteq \operatorname{Im} M$ 非平凡。

考虑 $v \in V$、$A \in \mathcal{A}$ 和 $R \in \mathcal{R}$。由于 $RA \in \mathcal{R}$，我们有 $RAv = 0$，因此 $Av \in \operatorname{Ker} R$。于是，$V$ 关于 $\mathcal{A}$ 不变。 □

例 8. 考虑例 5 中的系统。在例 6中，已证明该系统的雅可比代数是下三角矩阵的集合。与例 7 类似，我们发现该代数的根基为 $\begin{pmatrix} 0 & 0 \\ \lambda & 0 \end{pmatrix}$。该根基的公共核由第二个基向量张成，从而得到约化 $y' = -y + y^2$ (其中 $y = x_2$)。

定义 5（半单代数）。若一个代数 $\mathcal{A} \subseteq k^{n\times n}$ 的根基为零，则称其为半单。

我们将使用半单代数的以下刻画。

定理 3（Wedderburn-Artin，[27, 定理 2.4.3 和 2.6。2]）。设 $\mathcal{A} \subseteq k^{n\times n}$是一个半单代数。那么存在

1. 代数 $\mathcal{A}_1 \subseteq k^{n_1\times n_1}$，……，$\mathcal{A}_\ell \subseteq k^{n_\ell\times n_\ell}$，使得对每个 $1 \leqslant i \leqslant \ell$，$\mathcal{A}_i$ 在 k^{n_i} 中都没有非平凡真不变子空间；*2.* 整数 m_1，……，m_ℓ，使得 $n_1m_1 + \ldots + n_\ell m_\ell = n$；*3.* k^n中的一个基

使得在这个基下，我们有

$$\mathcal{A} = \left\{ \operatorname{Diag}(\underbrace{A_1, \ldots, A_1}_{m_1 \text{ times}}, \ldots, \underbrace{A_\ell, \ldots, A_\ell}_{m_\ell \text{ times}}) \mid A_1 \in \mathcal{A}_1, \ldots, A_\ell \in \mathcal{A}_\ell \right\}, \quad (5)$$

其中$\operatorname{Diag}(B_1, \ldots, B_N)$表示以 B_1，……，B_N为块的分块对角矩阵。

例 9。考虑所有以下形式的矩阵组成的集合

$$\begin{pmatrix} a & b & 0 & 0 \\ -b & a & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & c \end{pmatrix}, \quad \text{where } a, b, c \in \mathbb{Q}.$$

这是形如（5）的一个半单代数，其中有 $\ell = 2$、$m_1 = 1$和 $m_2 = 2$。

在定理3中的分解（5）里，当 $\ell = 1$ 和 $m_1 = 1$ 时， k^n 中不存在不变子空间；但若 $k \neq \overline{k}$，则 $\overline{k}^n$ 中仍可能存在不变子空间，其中 $\overline{k}$ 是域 k的代数闭包。这些子空间可以利用代数的中心求得。

定义 6（中心／中心化子）。设 $\mathcal{A} \subseteq k^{n\times n}$ 是一个代数。

- $\mathcal{A}$ 的中心是所有满足以下条件的 $M \in \mathcal{A}$ 的集合：对于每个$A \in \mathcal{A}$，都有 $MA = AM$。
- $\mathcal{A}$ 的中心化子是所有满足以下条件的 $M \in k^{n\times n}$ 的集合：对于每个 $A \in \mathcal{A}$，都有 $MA = AM$。

由于对于每个固定的 A， $AM = MA$ 都是关于各项的线性方程组，因此可以通过求解线性方程组来计算中心和中心化子。

引理 4。设 $\mathcal{A} \subseteq k^{n\times n}$ 是一个子代数。设 $\mathcal{C}$ 是 $\mathcal{A}$的中心化子。对于每个 $C \in \mathcal{C}$，C 的每个特征子空间都是$\overline{k}^n$中 $\mathcal{A}$ 的不变子空间。

证明。设 V 是 C 中对应特征值 λ的一个特征子空间。设$A \in \mathcal{A}$。那么，对于 $v \in V$，有 $C(Av) = (CA)v = (AC)v = \lambda Av$，因此 Av也属于 V。 □

引理 *5.* 设 $\mathcal{A} \subseteq \mathbb{Q}^{n\times n}$ 是一个半单代数。设 $M \in \mathcal{A}$ 是一个矩阵，且 M 的特征多项式具有 $p(t)^d$ 的形式，其中 $p(t)$ 是 $\mathbb{Q}$-不可约的。设 $\mathcal{Z}$ 和 $\mathcal{C}$ 分别是 $\mathcal{A}$ 的中心和中心化子。那么，等式 $\dim \mathcal{C} = d^2 \dim \mathcal{Z}$ 等价于：在 $\mathcal{A}$ 的 *Wedderburn- Artin* 分解 *(5)* 中，有 $\ell = 1$ 和 $m_1 = d$。

证明。考虑 $\mathcal{A}$ 的 Wedderburn-Artin 分解 (5)。对于每个$1 \leqslant i \leqslant \ell$，我们将 $\mathcal{A}_i$ 的中心记为 $\mathcal{Z}_i$。那么 $\dim \mathcal{Z} = \dim \mathcal{Z}_1 + \ldots + \dim \mathcal{Z}_\ell$。 $\mathcal{A}$ 中任意元素的特征多项式的不可约因子数至少为 $m_1 + \ldots + m_\ell$，因此 $d \geqslant m_1 + \ldots + m_\ell$。利用舒尔引理 [27, 定理 2.1.1] 直接计算可得， $\mathcal{A}$ 的中心化子 $\mathcal{C}$ 同构于 $\mathrm{Mat}_{m_1}(\mathcal{Z}_1) \times \ldots \times \mathrm{Mat}_{m_\ell}(\mathcal{Z}_\ell)$，其中 $\mathrm{Mat}_{m_i}(\mathcal{Z}_i)$ 表示由 $\mathcal{Z}_i$ 中的元素构成、每个块均为一个元素的 $m_i \times m_i$-块矩阵空间（参见 [27,定理 2.6.4]）。因此

$$\dim \mathcal{C} = m_1^2 \dim \mathcal{Z}_1 + m_2^2 \dim \mathcal{Z}_2 + \ldots + m_\ell^2 \dim \mathcal{Z}_\ell.$$

对右侧进行界定后，我们可以写成

$$\dim \mathcal{C} \leqslant (m_1 + \ldots + m_\ell)^2 (\dim \mathcal{Z}_1 + \ldots + \dim \mathcal{Z}_\ell) \leqslant d^2 \dim \mathcal{Z}$$

当且仅当 $\ell = 1$ 且 $d = m_1$时，两个不等式均取等号，这就证明了该引理。 □

算法 2 寻找代数的非平凡不变子空间

输入代数 $\mathcal{A} \subseteq \mathbb{Q}^{n\times n}$的一组基 $B_1, \ldots, B_N \in \mathbb{Q}^{n\times n}$；

输出 以下之一：

- 若 $\overline{\mathbb{Q}}^n$ 中不存在关于 $\mathcal{A}$不变的子空间，则返回否；
- $\mathbb{Q}^n$ 中关于 $\mathcal{A}$不变的非平凡真子空间；
- $\overline{\mathbb{Q}}^n$ 中关于 $\mathcal{A}$不变的子空间极大链。

考虑边界情况：

(步骤 1)如果 $N = n^2$，返回 NO。**(步骤 2)**对于任意非零向量 v，考虑由 B_1v、……、B_Nv 张成的空间 V。如果 dim $V < n$，**返回** V。

考察根基：

(步骤 3) 使用 Dixon 定理 [28, Theorem 11]，求出 $\mathcal{A}$ （定义 4）之根基 $\mathcal{R}$ 的一组基。(步骤 4) 如果 $\dim \mathcal{R} > 0$，计算 $\mathcal{R}$ 的基元素的公共核并返回该核（参见引理 2）。半单情形：(步骤 5) 令 $D := 1$。(步骤 6) 计算 $M := \sum_{i=1}^{N} a_i B_i$，其中 a_1、……、a_N 是从 $\{1, 2, \ldots, D\}$ 中独立且均匀随机抽取的。(步骤 7) 如果 M 的特征多项式至少有两个不同的 $\mathbb{Q}$-不可约因子（例如 $p_1(t)$ 和 $p_2(t)$）：(a) 检查 Ker $p_1(M)$ 关于 B_1、……、B_N 的不变性。(b) 如果它是不变的，返回 Ker $p_1(M)$。否则，令 $D := 2D$ 并转到（步骤 6）。

(步骤 8)将 M 的特征多项式写成 $p(t)^d$，其中 $p(t)$ 是 $\mathbb{Q}$-不可约的。

(步骤 9)计算 $\mathcal{Z}$ 的中心化子 $\mathcal{C}$ 和中心 $\mathcal{A}$ (定义 6)。

(步骤 10) 如果 维数$\mathcal{C} < d^2$ 维数 $\mathcal{Z}$，则令 $D := 2D$ 并 **转到 (步骤 6)**。

(步骤 11) 令 C_1,..., C_s 为 $\mathcal{C}$ 的一组基。令 $C := \sum_{i=1}^{s} b_i C_i$，其中 $b_1, \ldots, b_s$ 是从 $\{1, 2, \ldots, D\}$ 中独立且均匀随机抽取的。

(步骤 12)计算 $q(t)$，即 C 的最小多项式。如果 q 是 $\mathbb{Q}$-可约的，或$\deg q < d$ $\dim \mathcal{Z}$，则设 $D := 2D$ 并 **转到 (步骤 6)**。**(步骤 13)** 令 V_1，……，V_ℓ 14(其中 $\ell = d\dim \mathcal{Z}$) 为 C 的特征子空间。**(步骤 14) 返回** $V_1 \subset V_1 \oplus V_2 \subset \ldots \subset V_1 \oplus V_2 \oplus \ldots \oplus V_{\ell-1}$。

3.4. 不变子空间搜索：如何找到一个不变子空间

在本小节中，我们给出用于寻找不变子空间（如果存在）的算法 2 。本小节其余部分将用于证明该算法的正确性和终止性，见命题 2。

命题 1。 设 $\mathcal{A} \subseteq \mathbb{Q}^{n\times n}$ 是一个半单代数，且其中不存在非平凡真 $\mathcal{A}$-不变子空间。设 $M_1, .., M_N$ 是 $\mathcal{A}$ 作为向量空间的一组基。那么多项式

$$\det(x_1 M_1 + \ldots + x_N M_N) \in \mathbb{Q}[x_1, \ldots, x_N]$$

具有 P^d 的形式，其中 P 在 $\mathbb{Q}$ 上不可约。

注 4（基的重要性）。 虽然命题 1 的陈述听起来相当自然，但实际情况颇为微妙：若在命题陈述中用 的生成元集合替代线性基，结论便不再成立 [29, 定理 1.2 和第2节。1]。

命题 *1* 的证明。 通过对 $\mathbb{Q}$ 进行坐标变换，我们可以假设 M_1 是单位矩阵。

令 $\overline{\mathcal{A}}$ 为 $\mathcal{A}$ 的复化。根据韦德伯恩–阿廷定理 [27,推论 2.4，4], 存在 $n_1, \ldots, n_\ell$，使得 $N = n_1^2 + \ldots + n_\ell^2$，并且

$$\overline{\mathcal{A}} \cong \mathrm{Mat}_{n_1}(\mathbb{C}) \times \ldots \times \mathrm{Mat}_{n_\ell}(\mathbb{C}). \tag{6}$$

于是，原表示 $\mathbb{Q}^n$ 的复化 $\mathbb{C}^n$ 可根据 $\mathcal{A}$ 定理 2.6.2] 分解为 [27,

$$\mathbb{C}^n = k_1 V_1 \oplus k_2 V_2 \oplus \ldots \oplus k_\ell V_\ell, \tag{7}$$

其中 $V_i \cong \mathbb{C}^{n_i}$ 是 $\mathrm{Mat}_{n_i}(\mathbb{C})$ 的唯一不可约表示。我们将与 (7) 对应的基变换记为 $C \in \mathbb{C}^{n\times n}$。那么 CMC^{-1} 是分块对角矩阵，其中各块的维数如 (7) 所示。由于 $M_1, \ldots, M_N$ 张成整个 $\overline{\mathcal{A}}$，CMC^{-1} 的不同非零项是 $\mathbb{C}$-线性无关的线性形式，属于 $x_1, \ldots, x_N$。将这些形式记为 $y_1, \ldots, y_N$，可得到一个可逆矩阵 $B \in \mathbb{C}^{N\times N}$，使得 $\mathsf{y} := B\mathsf{x}$，并且在必要时对 $y_1, \ldots, y_N$ 重新排序后，有

$$CMC^{-1} = \mathrm{diag}(\underbrace{Y_1, \ldots, Y_1}_{k_1 \text{ times}}, \ldots, \underbrace{Y_\ell, \ldots, Y_\ell}_{k_\ell \text{ times}}),$$

其中，Y_i 是一个元素为 $y_{n_1+\ldots+n_{i-1}^2+1}$、……、$y_{n_1^2+\ldots+n_i^2}$（$1 \leqslant i \leqslant \ell$）的矩阵。

于是我们有

$$\det(M) = \det(CMC^{-1}) = \det(Y_1)^{k_1} \ldots \det(Y_\ell)^{k_\ell}.$$

此外，由于 M_1 是单位矩阵，$\det(M)|_{x_1=x_1+t}$ 作为关于 t 的多项式，是 $-M$的特征多项式。令 $Q(\mathsf{x}) := \det Y_1 \ldots \det Y_\ell \in \mathbb{Q}[\mathsf{x}]$。那么，$Q|_{x_1=x_1+t}$ 作为关于 t 的多项式，是 $-M$的最小多项式。

由于 $\det Y_i$ 是一个独立元素矩阵的行列式，因此它在 $\mathbb{C}$上不可约。令 p (x) 为 $\det M$ 的一个 $\mathbb{Q}$-不可约因子。那么 p 整除Q，因此，如有必要，对 Y_i 进行重新排序后，我们可以假设对于 $r \leqslant \ell$， p(x) $= \det Y_1 \dots \det Y_r$。假设 $r < \ell$。设 $p_0(t) := p(x_1 - t, x_2, \dots, x_N)$，并考虑$p_0(M)$。我们将得到

$$Cp_0(M)C^{-1} = \operatorname{diag}(\underbrace{0, \dots, 0}_{k_1+\dots+k_r \text{ times}}, \underbrace{p_0(Y_{r+1}), \dots, p_0(Y_{r+1})}_{k_{r+1} \text{ times}}, \dots, \underbrace{p_0(Y_{r+1}), \dots, p_0(Y_{r+1})}_{k_\ell \text{ times}}).$$

由于 p_0 与 Y_{r+1}、……、Y_ℓ的特征多项式互素，因此矩阵 $p_0(Y_{r+1})$、……、p_0 (Y_ℓ) 均为非奇异矩阵。因此，$Cp_0(M)C^{-1}$ 的核恰好是前 $k_1 + \dots + k_r$ 个基向量的张成空间。因此， $p_0(M)$的核是 C^{-1}的最前面相同数量列的张成空间。因此， $p_0(M)$的核是 $\overline{\mathcal{A}}$-不变的，且定义在 $\mathbb{C}$上。另一方面， $p_0(M)$的元素属于 $\mathbb{Q}$ (x)，所以 $p_0(M)$的核实际上定义在 $\mathbb{C} \cap \mathbb{Q}$(x) $= \mathbb{Q}$上。因此， $p_0(M)$的核给出了 $\mathbb{Q}^n$ 的一个非平凡$\mathcal{A}$-不变子空间，这与该表示的不可约性矛盾。因此， p 必定等于 Q，所以 $\det M$ 必定是 p的幂。 □

该命题的证明提供了一种求取 $\deg P$ 的次数的方法。

推论 3。采用命题 1 的证明中的记号（见（6）），$\deg P = n_1 + n_2 + \dots + n_\ell$。

命题 2。*算法 2 是正确的，并且以概率为 1 终止。*

注 5（关于终止概率）。所谓“以概率为 1 终止”，是指该算法在一个无限概率空间中进行随机选择，并且除概率为零的一个集合外，对于所有选择都会终止。简单地说，该算法会反复抛硬币；只要至少出现一次正面，它就会终止，而这最终会以概率为 1 发生。

命题 2 的证明。我们首先证明其正确性。如果算法在**(步骤 1)**处返回，则 $\mathcal{A}$ 是全矩阵代数，且不具有任何非平凡的真不变子空间。如果算法在**(步骤 2)**处返回，则返回的子空间按构造是不变的。如果算法在**(步骤 4)**处返回，则返回的子空间是非零的、真子空间（因为如果 $\mathcal{A}$ 是零代数，算法本会在**(步骤 2)**处返回），并且根据引理 2 是不变的。

还需考虑算法在**(步骤 5)**后返回的情况。如果算法在**(步骤 7)**(b)处返回，那么返回的子空间按构造是不变的，并且非零，因为 $p_1(t)$ 整除 M的特征多项式，所以$p_1(M)$ 是一个奇异矩阵。最后，考虑算法在**(步骤 14)**处返回的情况。考虑定理3中的分解（5），其中 $\mathcal{A}$。如果算法到达了**(步骤 11)**，则其中包含一个特征多项式为

$p(t)^d$ 具有 $\mathbb{Q}$-不可约的 $p(t)$，且 $\dim \mathcal{C} = d^2 \dim \mathcal{Z}$。引理 5 表明，在分解（5）中，我们有 $\ell = 1$ 和 $m_1 = d$。因此，整个空间 $\mathbb{Q}^n$可写为 $U_1 \oplus U_2 \oplus \ldots \oplus U_d$，其中每个 U_i 都是 $\mathcal{A}$-不变的，且没有真非平凡的 $\mathcal{A}$-不变子空间。[27, 推论 2.2.4] 表明，在$\overline{\mathbb{Q}}$上，每个 U_i 都可以分解为至多 $\dim \mathcal{Z}$ $\mathcal{A}$ 个不变子空间的直和。因此，根据 [27, 定理 2.6.2]，整个 $\overline{\mathbb{Q}}^n$ 可以分解为至多 $d\dim\ \mathcal{Z}\mathcal{A}$-不变子空间。

引理 4 蕴含了以下结论：对于 V_i 在 **（步骤 13）** 中的每一个，都是关于 $\mathcal{A}$ 的不变子空间。由于它们共有 $d\dim\ \mathcal{Z}$ 个，因此每个 V_i 都不包含非平凡的真 $\mathcal{A}$-不变子空间。因此，在 **（步骤 14）** 返回的链 $V_1 \subset V_1 \oplus V_2 \subset \ldots \subset V_1 \oplus V_2 \oplus \ldots \oplus V_{\ell-1}$ 是极大的。这就完成了对该算法正确性的证明。

现在我们将证明该算法以概率为 1 终止。考虑定理 3 中 $\mathcal{A}$ 的分解 (5)。考虑变量 z_1、.....、z_N以及矩阵 $M_0 := \sum_{i=1}^{N} z_i B_i$。那么，（步骤 6）处的 M 是 $z_i = a_i$ 处 M_0 的一个特化。令 $P(z_1, \ldots..、 z_N, t)$ 为 M_0 的特征多项式。考虑 $\mathcal{A}$ 的分解 (5)。对于每个 $1 \leqslant i \leqslant \ell$，我们将命题 1 应用于对应于 $\mathcal{A}_i$ 的块，并得到一个 $\mathbb{Q}$-不可约的 P_i 及其幂 d_i。因此，我们得到 M_0 的以下因式分解

$$P = P_1^{d_1 m_1} P_2^{d_2 m_2} \ldots P_\ell^{d_\ell m_\ell}.$$

矩阵的特征多项式 M在 **(步骤 6)** 处计算得到 $P(a_1、 \ldots..、 a_N、 t)$。假设 $P_i(a_1、 \ldots..、 a_N、 t)$ 对每个 $\mathbb{Q}$ 都是 -可约的，$1 \leqslant i \leqslant s$ 且这些多项式彼此不同。

- 假设 $\ell > 1$。那么，$p_1(t)$ 在**(步骤 7)** 中得到的 $P_i(a_1、 \ldots..、 a_N、 t)$对于某个 i 将等于核 $p_1(M)$ 将是对应于分解中第i块的子空间（5）。该子空间是不变子空间，因此会在 **(步骤 7)**(b) 中返回。

- 假设 $\ell = 1$。我们将以与上文研究 M 类似的方式研究矩阵 C。设 $y_1、 \ldots..、 y_s$ 为独立变量，并定义 $C_0 := y_1 C_1 + \ldots + y_s C_s$。由引理 5 证明中的相同论证可知，对于某个整数 r，有 $\mathcal{C} \cong \mathrm{Mat}_r(\mathcal{Z})$。根据 [27, 命题 2.3.4], , 代数 $\mathcal{C}$ 是单代数，且每个 $\mathcal{C}$-模（特别是我们的环境空间 $\mathbb{Q}^n$）都是同一个 $\mathcal{C}$-模的若干个同构副本的直和。我们对这个模应用命题 1，推得 C_0 的特征多项式具有 $Q(y_1, \ldots..., y_s, t)^h$ 的形式，其中 h 是某个整数，Q 是 $\mathbb{Q}$-不可约多项式。此外，由推论 3，$\deg Q = d \dim\ \mathcal{Z}$。假设 $Q(b_1, \ldots..., b_s, t)$ 是 $\mathbb{Q}$-不可约的。那么 $Q(b_1, \ldots..., b_s, t)$ 将是 C 的最小多项式，因此该多项式不满足（步骤 12）的条件，算法也就会终止，而不会返回（步骤 6）。

结合上文中两个带下划线的假设可知，若满足以下条件，算法将返回固定值 D：

1. 对每个 $1 \leqslant i \leqslant s$， $P_i(a_1, \ldots, a_N, t)$ 都是 $\mathbb{Q}$-可约的，且这些多项式彼此不同；2. $Q(b_1, \ldots, b_s, t)$ 是 $\mathbb{Q}$-不可约的。

[30, 定理 2.1] 意味着存在常数 C_0、C_1，使得当 $D > C_0$ 时，任意 $P_i(a_1, \ldots, a_N, t)$ 和 $Q(b_1, \ldots, b_s, t)$ 可被 $\mathbb{Q}$-约化的概率小于 $\frac{C_1}{\sqrt[3]{D}}$。此外，施瓦茨-齐佩尔引理 [31, 命题 98] 意味着存在常数 C_2，使得任意 $P_i(a_1, \ldots, a_N, t)$ 相等的概率不超过 $\frac{C_2}{D}$。因此，对于 $D > C_0$， D 将被更新的概率至多为 $\frac{C_1}{\sqrt[3]{D}} + \frac{C_2}{D}$。这个数最终会小于 0.99，因此不终止的概率将被限制在 $0.99 \cdot 0.99 \cdot \ldots = 0$以内。

□

3.5. 不变子空间搜索：如何找到一条链

本节将介绍如何递归地使用算法 2，以求得 $\overline{\mathbb{Q}}$ 关于常微分方程组雅可比代数$\mathcal{A} \subset \mathbb{Q}^{n\times n}$ 的极大不变子空间链。我们将 $\mathcal{A}$ 的一组基记为 $B_1, \ldots, B_N$

当算法 2 应用于 $B_1, \ldots, B_N$ 时，若返回 NO 或极大不变子空间链，则问题已解决。因此，我们考虑算法 2 返回单个不变子空间 $V \subset \mathbb{Q}^n$的情形。在这种情况下，我们考虑两个子问题：

1. 限制。由于 V 关于 $B_1, \ldots, B_N$ 不变，因此存在定义良好的限制 $B_1|_V, \ldots, B_N|_V$。我们在 V 中固定一组基，并将这些受限算子的矩阵表示也记为 $B_1^*, \ldots, B_N^*$。 2. 商空间。考虑商空间 $\mathbb{Q}^n/V$ 和商映射π: $\mathbb{Q}^n \to \mathbb{Q}^n/V$ (参见 [32, 3.83, 3.88])。由于 V 关于$B_1, \ldots, B_N$ 不变，我们可以考虑商算子 $B_1/V, \ldots, B_N/V$(参见 [32, 5.14])，并将其矩阵表示记为 $B_1^\circ, \ldots, B_N^\circ$。注意，对于它们的每个公共不变子空间 $U \subset \mathbb{Q}^n/V$，子空间 $\pi^{-1}(U) \subset \mathbb{Q}^n$ 关于 $B_1, \ldots, B_N$ 不变。

注意，上述矩阵表示可以通过求解含有 n 个变量的线性方程组来计算。因此，我们可以递归地处理代数$\langle B_1^*$、.....、$B_N^*\rangle$ 在 V上的情形，以及 $\langle B_1^\circ$、.....、$B_N^\circ\rangle$ 在 $\mathbb{Q}^n/V$上的情形。如果得到的不变子空间极大链是

$$0 \subsetneq V_1 \subsetneq \ldots \subsetneq V_s \subsetneq V \quad \text{and} \quad 0 \subsetneq U_1 \subsetneq \ldots \subsetneq U_r \subsetneq \mathbb{Q}^n/V,$$

然后，我们可以返回以下适用于B_1、.....、B_N的极大不变子空间链

$$0 \subsetneq V_1 \subsetneq \ldots \subsetneq V_s \subsetneq V \subsetneq \pi^{-1}(U_1) \subsetneq \ldots \subsetneq \pi^{-1}(U_r) \subsetneq \mathbb{Q}^n.$$

3.6. 汇总

本节将前面各节中的子程序汇总起来，形成用于寻找最大集总链的完整算法。

算法 3 寻找极大集总链 输入：常微分方程组 $\mathbf{x}' = \mathbf{f}(\mathbf{x})$，其中 $\mathbf{x} = (x_1, \ldots, x_n)$，$\mathbf{f} = (f_1, \ldots, f_n) \in \mathbb{Q}[\mathbf{x}]^n$；输出：极大集总链（参见定义 2 和例 4）；（步骤 1）计算 $\mathbf{f}$ 的雅可比矩阵 $J(\mathbf{x})$，以及其分解（如式（4）所示）得到的矩阵 $J_1, \ldots, J_\ell \in \mathbb{Q}^{n\times n}$。（步骤 2）使用算法 1 计算该系统的雅可比代数 $\mathcal{A} = \langle I_n, J_1, \ldots, J_\ell \rangle$ 的基 $B_1, \ldots, B_N$。（步骤 3）按照第 3.5 节所述递归应用算法 2，计算 $\overline{\mathbb{Q}}^n$ 中关于 $\mathcal{A}$ 不变的子空间的极大链 $V_1 \subsetneq \ldots \subsetneq V_s$。（步骤 4）对于每个 $1 \leqslant i \leqslant s$，求一个矩阵 L_i，其列构成 V_i 的一组基。（步骤 5）返回 $L_1, \ldots, L_s$。

---

4. 实现与性能

我们已使用 Julia 语言实现算法3 （以及它所依赖的所有算法） [33]，作为 `ExactODEReduction.jl`软件包的一部分。该软件包及复现我们结果所需的相关资源均可在以下地址免费获取：

https://github.com/x3042/ExactODEReduction.jl

我们使用 AbstractAlgebra.jl 和 Nemo.jl 库 [34]。在内部，这意味着使用 FLINT [35] 和 Calcium [36] （用于复数运算）。我们使用了来自 [22] 的代码版本，以提高计算所得集总结果的可解释性。此外，在开发阶段，我们对软件包的各个组件进行了性能分析，测试对象是 SuiteSparse 数据集 [37]中的稀疏矩阵集合。我们的实现支持手动输入模型，也支持从 ERODE *.ode 格式文件导入模型 [3, 第 3 节2]。我们提供文档、安装说明和使用示例。

注 6（编码标量参数）。文献中的模型通常包含标量参数。我们的算法将每个此类参数 k 变换为一个状态 $k(t)$，使其满足方程 $k'(t) = 0$ （同样的变换在 ERODE 中称为“柯里化”[38, 第 15] 页）。这样一来，也可以在参数空间中寻找约化，而不仅限于状态空间。另一种

处理参数的方法是将其并入系数域（CLUE 允许这样做），从而允许约化的系数中包含参数。目前尚未实现此功能，但本文介绍的大部分理论和算法均可用于这种情况。

我们将在一组基准测试上展示我们实现的性能[2]。我们使用来自 BioModels 数据库的基准测试 [21]，收集于 [39]，其维数范围从4 到133。我们对每个模型在有理数域上运行算法3 。表1 包含按模型维数汇总的基准测试结果。对于每个范围，我们报告：

- 所考虑的模型数量；
- 所找到的约化链的（平均）长度；
- 不等价约化的（平均）数量，其中等价性按加入具有恒定动力学的状态来判定。我们选择报告这一项，是因为我们认为它可以合理地初步近似有意义的约化数量；
- 我们实现的（最小、平均、最大）运行时间；

模型信息		约化		运行时间（秒）		
维数	# 模型	#总计	# 非等价	Min.	平均值	Max.
2 - 9	44	4.02	1.39	0.0 秒	0.6 秒	0.66 秒
10 - 19	41	8.15	2.61	0.01 秒	0.21 秒	1.46 秒
20 - 29	46	9.65	2.13	0.08 秒	0.44 秒	1.48 秒
30 - 39	17	19.41	2.71	0.33 秒	1.74 秒	5.91 秒
40 - 59	25	29.08	6.08	0.78 秒	4.58 秒	26.71 秒
60 - 79	20	37.25	6.95	7.7 秒	34.57 秒	102.92 秒
80 - 99	11	42.91	7.09	24.46 秒	96.38 秒	497.26 秒
100 - 133	4	89.0	21.5	75.15 秒	202.52 秒	312.02 秒

表 1：按模型维度汇总的基准测试结果

测试耗时是在一台配备 2 个 1.60GHz 核心和 8 Gb 随机存取存储器的笔记本电脑上测得的[3]。值得注意的是，在所考虑的 208 个模型中，202 个模型至少找到了一种约化，其中 154 个模型可进行非恒定约化。

表中的计时不包括正化步骤 [22],的开销，该步骤是可选的。此处，我们的算法使用 Polymake [40] 库。借助

[2]模型可在此处获取：https://github.com/x3042/ExactODEReduction.jl/tree/main/data/ODEs，提交哈希 678d32c5bbc8beedc9e22b673238cde0ec673a46。[3]完整表格请参阅 https://github.com/x3042/ExactODEReduction.jl/blob/main/benchmark/biomodels_基准测
提交哈希 23c9f532aa316cbef59a8e3e6be04156a3d9c3eb。

正化步骤，在大多数情况下，运行时间最多增加一倍，而且通常完全无法察觉到增幅[4]。在较早版本的算法3实现中，在**(步骤 2)** 计算数据集上的代数基往往是明显的瓶颈。按照第1 节3.2所述对算法进行修改后，目前最耗时的步骤是在**(步骤 3)** 算法3中执行限制和商化过程。求解多个线性方程组以求得受限算子和商算子的矩阵表示，是此处明显的瓶颈。

5. 案例研究

5.1. 因子 *Va* 的失活

我们将考虑一个模型，该模型出自[41] BioModels 数据库 [21]，编号为 BIOMD0000000365。因子 V 是参与凝血过程（使血液由液态转变为凝胶状）的蛋白质，因此与血管修复和血栓形成密切相关。具体而言，它有助于活化抗凝血蛋白 C。活化因子 V（即因子 Va）则不再具有这一作用。 [41]提出并研究了一个模型，用于描述活化蛋白 C（APC）使因子 Va 失活的过程。

因子 Va 由重链（HC）和轻链（LC）组成，APC 通过轻链与其结合。该模型包含以下物种

- 因子 Va 及其变体 Va_3、Va_5、Va_6、Va_{53}、Va_{56}、Va_{36}和 Va_{536}；
- LC、HC，以及后者的变体（HC_3、HC_5等），它们分别对应 Va 的各个变体；
- 因子 Va 的 A_1 结构域、$Va_{LC\cdot A1}$ 以及 A_2 结构域的各种形式，例如 Va_{A3}、Va_{A53} 等。
- APC、由其形成的复合物以及 LC/Va（例如 APC·Va_3）。

总计而言，该模型包含 30 个变量和 9 个参数，并且如备注 6 所述，参数被编码为常数状态。我们的代码能在不到 5 秒的时间内，在笔记本电脑上找到长度为 14 的极大集总链。具有非零动力学的最小约化维数为三，并涉及两个参数（与例 4 中的情况类似）：

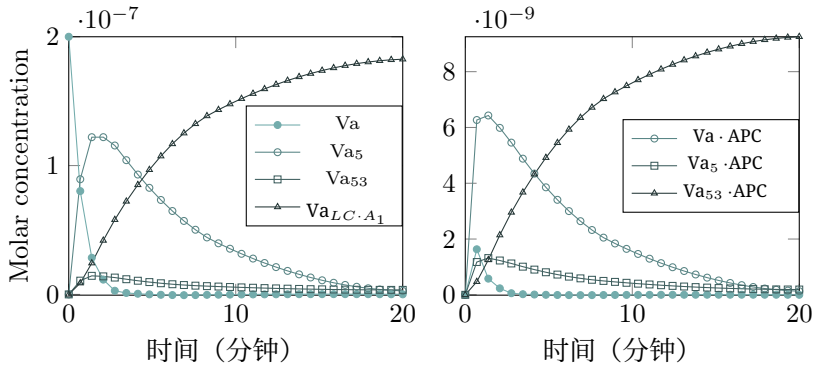
$$\begin{cases} y_1' = -k_1 y_1 y_2 + k_2 y_3, \\ y_2' = -k_1 y_1 y_2 + k_2 y_3, \\ y_3' = k_1 y_1 y_2 - k_2 y_3. \end{cases} \tag{8}$$

[4]一个值得注意的例外是那些能够进行大幅约化且系数较大的模型。例如，维数为 76 的模型 BIOMD0000000153 有 22 个非平凡约化，其维数均为 55 或更高；应用正化例程会使总运行时间从 40 秒增加到 1240 秒。

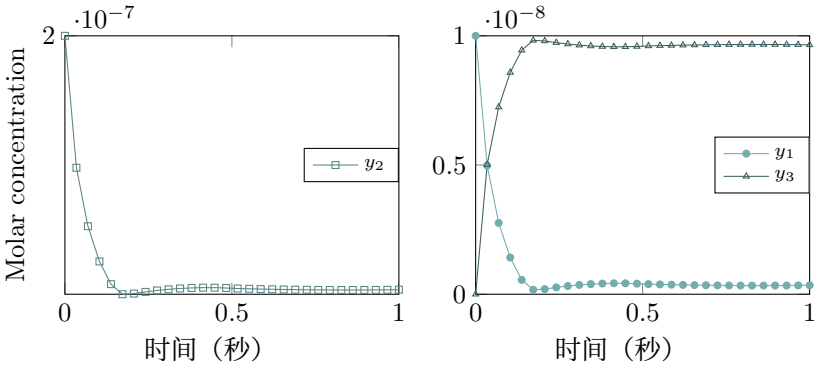
宏变量为

$y_1 = [\text{APC}], y_2 = [\text{轻链}] + [\text{Va}] + [\text{Va}_3] + [\text{Va}_{36}] + [\text{Va}_5] + [\text{Va}_{53}] + [\text{Va}_{536}] + [\text{Va}_{56}] + [\text{Va}_{\text{LC}\cdot A_1}]$, $y_3 = [\text{轻链}\cdot\text{APC}] + [\text{Va}\cdot\text{APC}] + [\text{Va}_3\cdot\text{APC}] + [\text{Va}_{36}\cdot\text{APC}] + [\text{Va}_5\cdot\text{APC}] + [\text{Va}_{53}\cdot\text{APC}] + [\text{Va}_{536}\cdot\text{APC}] + [\text{Va}_{56}\cdot\text{APC}] + [\text{Va}_{\text{LC}\cdot A_1}\cdot\text{APC}]$.

变量 y_2 （分别为 y_3）可表示为未与APC结合（分别为已与APC结合）的轻链总浓度。因此，约化（8）聚焦于APC与轻链的结合/解离过程。结果表明，其他过程（例如重链与轻链之间的反应）变得无关紧要；特别是，重链（HC）$_n$ 物种完全不会出现在宏变量中。



(a) 原始模型中出现在 y_2 和 y_3中的部分状态



(b) 约化模型的状态

图2：模型的数值模拟：[41] 及其约化，采用来自 [41]

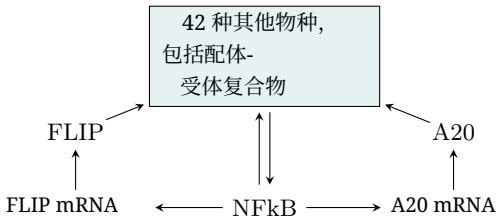


图3：相关化学物种及其相互依赖关系

从数值角度来看，[5], 约化（8）可解释为精确的时间尺度分离，因为宏变量的动力学相较于原系统的动力学表现为瞬态。更确切地说，文献 [41] 研究了原系统，发现其在 1200 秒的时间跨度内具有非平凡动力学。特别是，对宏变量 y_2有贡献的变量便是如此，见图 2a。另一方面，如图 2b 所示，宏变量 y_1、y_2、 y_3 的动力学快得多，并在不到一秒后达到稳态。

5.2. 细胞死亡模型

在本小节中，我们考虑一个在 [44] 中设计的模型，用以研究细胞凋亡（程序性细胞死亡）对肿瘤坏死因子（TNF）刺激的敏感性。该模型总体上涉及 47 种化学物种及它们之间的大量相互作用，如 [44, 图 1]所示。我们的代码生成了一条长度为 23 的极大集总链（其中 16 个具有非恒定动力学）。

我们将考虑维数最小的非恒定约化。它涉及两种蛋白质 A20 和 FLIP，其浓度取决于相应信使RNA A20_mRNA 和 FLIP_mRNA 的浓度。这些信使RNA的浓度由核内 NF-κB（NFκB_N）的浓度决定。后者（直接或间接）取决于许多其他物种，包括前述蛋白质 A20。

这些物种及其相互关系总结见图 3，相应的微分方程如下：

$$[A20]' = k_1[A20_mRNA] + k_2, \qquad [A20_mRNA]' = k_5[NF\kappa B_N],$$
$$[FLIP]' = k_3[FLIP_mRNA] + k_4, \quad [FLIP_mRNA]' = k_6[NF\kappa B_N],$$

其中 $k_1, .., k_6$ 是数值参数。我们的代码找到了一个三维约化，可以进一步直接简化为二维约化

[5]本文中的所有数值模拟均使用 ModelingToolkit [42] 和 DifferentialEquations.jl [43]完成

具有以下宏变量 y_1、y_2 以及约化后的系统：

$$\begin{cases} y_1 = \frac{k_6}{k_1}[A20] - \frac{k_5}{k_3}[FLIP], \\ y_2 = k_6[A20_mRNA] - k_5[FLIP_mRNA] \end{cases} \implies \begin{cases} y_1' = y_2 + \frac{k_2 k_6}{k_1} - \frac{k_4 k_5}{k_3}, \\ y_2' = 0 \end{cases}$$

因此，我们的思路是，尽管 A20 和 FLIP 都参与了一个复杂反应网络，但通过消除对 NFκB 的依赖，可以找到它们的一个线性组合，使其满足一个可显式求解的简单微分方程系统。这样的状态显式关系例如可以与微分不等式方法结合，以获得更紧的可达性界 [45]。

沿着约化链继续深入，可以逐步将更多物种纳入约化模型。例如，可以用类似的方法纳入 RIP蛋白与过渡受体的组合。

6. 结论

我们提出了一种新算法，该算法以常微分方程系统为输入，并生成该系统可能的最长精确线性约化链，使链中的每次约化都是前一次约化的细化。与现有工具相比，这一规范更加通用，因为它对新变量没有任何限制，只要求它们是原始变量的线性组合；同时也不需要指定初始可观测量或进行猜测。我们预计，可以借鉴 [16], 中的思路，将我们的方法扩展到右端为有理函数的系统；这将留待未来研究。

我们提供了一个公开可用的 Julia 实现。在普通硬件上，我们的代码只需几分钟就能分析维数超过一百的模型。我们还展示了该方法在生命科学领域模型中的适用性。通过进一步改进，性能还可以提升。例如，可以先搜索线性第一积分（它们经常出现，例如在化学反应网络中），再利用这些常数约化跳过算法中的部分步骤。

由于所得约化是精确的，我们的工具可用于形式化验证，也可作为近似约化技术的预处理步骤。因此，精确性虽是一项重要特性，但也可视为一种局限，因为有些模型只有少数（甚至没有）精确约简。因此，未来研究的一个有趣方向是对我们的算法进行放宽，以寻找近似集总并给出严格的误差界。关于近似集总及基于奇异摄动理论的相关方法的现有成果，请参见 [46, 47, 48] 及其中的参考文献。有趣的是，我们算法的核心线性代数问题——寻找公共不变子空间——最近已从近似而严格的计算角度得到研究， [49, 50] 中的工作旨在对线性微分算子进行因式分解。我们期待这些论文中的思路同样适用于我们的研究情境。

致谢

我们要感谢 David E. Speyer 提供清晰详尽的说明[20]。我们要感谢 Mirco Tribastone 进行的有益讨论，也感谢 Rongwei Yang 就命题 1展开的讨论。我们要感谢审稿人提供的宝贵意见。GP 获巴黎法兰西岛大区（“XOR”项目）资助，并部分获得美国国家科学基金会资助，项目编号为 DMS-1853482、DMS-1760448 和 DMS-1853650。AD 获马克斯·普朗克信息学研究所资助。

参考文献

[1]J. Feret、V. Danos、J. Krivine、R. Harmer、W. Fontana，分子系统的内部粗粒化，《美国国家科学院院刊》106 (16) (2009) 6453–6458。网址：http://dx.doi.org/10.1073/pnas.0809908106

[2]J. Feret、T. Henzinger、H. Koeppl、T. Petrov，基于规则的系统的可聚合性抽象，《理论计算机科学》431 (2012)137–164。网址：https://doi.org/10.1016/j.tcs.2011.12.059

[3]L. Cardelli、M. Tribastone、M. Tschaikowski、A. Vandin，ERODE：用于常微分方程评估与约化的工具，载于：TACAS 2017，《计算机科学讲义》第 10206 卷，2017 年。网址：https://doi.org/10.1007/978-3-662-54580-5_19

[4]L. Cardelli, M. Tribastone, M. Tschaikowski, A. Vandin，多项式动力系统的极大聚合，《美国国家科学院院刊》114 (38) (2017) 10029–10034。网址：https://www.pnas.org/content/114/38/10029

[5]A. Ovchinnikov, I. P. Verona, G. Pogudin, M. Tribastone，CLUE：通过微分方程的约束聚合对动力学模型进行精确的极大约化，《生物信息学》（2021年）。网址：https://doi.org/10.1093/bioinformatics/btab010

[6] A. Antoulas，《大规模动力系统的近似》，设计与控制进展，工业与应用数学学会，2005年。

[7] E. Hubert, G. Labahn，动力系统的缩放不变量与对称性约化，《计算数学基础》13 (4) (2013年) 479–516。网址：https://doi.org/10.1007/s10208-013-9165-9

[8] F. Camporesi, J.Feret, 基于规则的模型的形式化约化,《理论计算机科学电子笔记》276 (2011) 29–59. URL https://doi.org/10.1016/j.entcs.2011.09.014

[9] P. J. Olver,等价、不变量与对称性, 剑桥大学出版社, 1995. URL https://doi.org/10.1017/cbo9780511609565[10] S. Sankaranarayanan,使用 c 自动抽象非线性系统

基变换

载于：第14届混合系统：计算与控制国际会议论文集，2011年。URL https://doi.org/10.1145/1967701.1967723[11]J. R. Faeder、M. L. Blinov、W. S. Hlavacek，信号转导网络的图形化规则表示，载于：2005年ACM应用计算研讨会论文集，2005年。URL https://doi.org/10.1145/1066677.1066712[12]V. Danos、J. Feret、W. Fontana、R. Harmer、J. Krivine，细胞信号传导的基于规则的建模，载于：CONCUR 2007——并发理论，2007年，第17–41页。URL https://doi.org/10.1007/978-3-540-74407-8_3[13]F. Camporesi、J. Feret、J. Hayman，上下文敏感流分析：模型约化层级，载于：系统生物学计算方法，2013年，第220–233页。URL https://doi.org/10.1007/978-3-642-40708-6_17 [14] F. Camporesi、J. Feret、K. Q. Lý，KaDE：将Kappa规则编译为（约化的）常微分方程模型的工具，载于：第十五届静态分析与系统生物学国际研讨会（SASB'17），《计算机科学讲义》/LNBI第10545卷，施普林格，2017年。[15]L. Cardelli、M. Tribastone、M. Tschaikowski、A. Vandin，微分等价的符号计算，《理论计算机科学》777（2019）132–154。URL https://doi.org/10.1016/j.tcs.2019.03.018[16]A. Jiménez-Pastor、J. P. Jacob、G. Pogudin，有理动力系统的精确线性约化，载于：系统生物学计算方法，施普林格国际出版，2022年，第198–216页。URL https://doi.org/10.1007/978-3-031-15034-0_10[17] G. Li、H. Rabitz，化学动力学中精确集总的一般分析，《化学工程科学》44（6）（1989）1413–1430。URL https://doi.org/10.1016/0009-2509(89)85014-6[18] L. Rónyai，有限代数结构的计算，《符号计算杂志》9（3）（1990）355–373。URL https://doi.org/10.1016/S0747-7171(08)80017-X

[19]A. Chistov、G. Ivanyos、M. Karpinski，有限维代数上模的多项式时间算法，载于：1997年国际符号与代数计算研讨会论文集，1997年，第68–74页。URL https://doi.org/10.1145/258726.258751

[20] D. Speyer，对“有没有一种简洁的方法来提取一组矩阵的不变子空间？”的回应
URL https://mathematica.stackexchange.com/questions/6519/is-there-a-clean-way-to-extrac

[21]R. S. Malik-Sheriff、M. Glont、T. V. N. Nguyen、K. Tiwari、M. G. Roberts、A. Xavier、M. T. Vu、J. Men、M. Maire、S. Kananathan、E. L. Fairbanks、J. P. Meyer、C. Arankalle、T. M. Varusai、V. Knight-Schrijver、L. Li、C. Dueñas-Roca、G. Dass、S. M. Keat- ing、Y. M. Park、N. Buso、N. Rodriguez、M. Hucka、H. Hermjakob，BioModels——共享生命科学计算模型15周年，《核酸研究》48 (D1) (2020) D407–D415。网址 https://doi.org/10.1093/nar/gkz1055[22] G. Pogudin、X. Zhang，基于正性的可解释精确线性约化，载于：E. Cinquemani、L. Paulevé（编），《系统生物学计算方法》，2021年，第91–107页。网址 https://doi.org/10.1007/978-3-030-85633-5_6[23]S. Dunn、A. Constantinides、P. Moghe，《生物医学工程数值方法》，学术出版社，2006年。网址 https://doi.org/10.1016/B978-0-12-186031-8.X5000-6[24] M. Feinberg，《化学反应网络理论基础》，Springer Cham，2019年。网址 https://doi.org/10.1007/978-3-030-03858-8[25] M. Reineke，每个射影簇都是一个箭图格拉斯曼簇，《代数与表示理论》16 (2013) 1313–1314。网址 https://doi.org/10.1007/s10468-012-9357-z[26]E. Nijholt、B. W. Rink、S. Schwenker，动力系统中的箭图表示与降维，《SIAM应用动力系统杂志》19 (4) (2020) 2428–2468。网址 https://doi.org/10.1137/20m1345670[27] Y. A. Drozd、V. V. Kirichenko，《有限维代数》，施普林格出版社，1994年。[28] M. R. Bremner，如何计算有限维代数的韦德伯恩分解

-维结合代数

《群、复杂性与密码学》3 (1) (2011) 47–66。网址 https://doi.org/10.1515/gcc.2011.003[29] I. Klep、J. Volčič，《群表示、行列式超曲面及其量子化》，载于：M. A. Bastos、L. Castro、A. Y. Karlovich

（编），《算子理论、泛函分析及其应用》，施普林格国际出版，2021年，第393–402页。[30] S. D. Cohen，伽罗瓦群的分布与希尔伯特不可约性定理，《伦敦数学会会刊》s3-43 (2) (1981) 227–250。网址 https://doi.org/10.1112/plms/s3-43.2.227[31] R. Zippel，有效多项式计算，施普林格，1993年。网址 http://dx.doi.org/10.1007/978-1-4615-3188-3[32] S. Axler，《正确地做线性代数》，Springer Cham，2015年。网址 https://doi.org/10.1007/978-3-319-11080-6[33]J. Bezanson、A. Edelman、S. Karpinski、V. B. Shah，Julia：一种数值计算的新方法，《SIAM评论》59 (1) (2017) 65–98。网址 https://doi.org/10.1137/141000671[34]C. Fieker、W. Hart、T. Hofmann、F. Johansson，Nemo/hecke：面向Julia编程语言的计算机代数与数论软件包，载于：2017年美国计算机协会国际符号与代数计算研讨会论文集，ISSAC '17，美国计算机协会，美国纽约州纽约，2017年，第157–164页。网址 https://doi.acm.org/10.1145/3087604.3087611[35] W. B. Hart，《快速数论库：简介》，载于：第三届国际数学软件大会论文集，ICMS'10，施普林格出版社，柏林、海德堡，2010年，第88–91页，https://flintlib.org。[36] F. Johansson，Calcium，载于：2021年国际符号与代数计算研讨会论文集，2021年。网址 https://doi.org/10.1145/3452143.3465513[37] T. A. Davis，算法1000：SuiteSparse:GraphBLAS：稀疏线性代数语言中的图算法

线性代

《美国计算机协会数学软件汇刊》45 (4)（2019年12月）。网址 https://doi.org/10.1145/3322125[38] ERODE：随机反应网络与微分方程的评估和约化，用户手册（2020年）。网址 https://www.erode.eu/docs/ERODE-manual.pdf[39] I. C. Pérez-Verona、M. Tribastone、A. Vandin，《BioModels知识库中精确模型约化的大规模评估》，载于：L. Bortolussi、G. Sanguinetti（编），《系统生物学计算方法》，施普林格国际出版，2019年，248–265页。[40] B. Assarf、E. Gawrilow、K. Herr、M. Joswig、B. Lorenz、A. Paffenholz、T. Rehn，使用Polymake计算凸包并统计整数点，《数学规划计算》9 (1)（2017年）1–38页。网址 http://dx.doi.org/10.1007/s12532-016-0104-z

[41]M. F. Hockin、K. M. Cawthern、M. Kalafatis、K. G. Mann，一个描述 APC 使因子 Va 失活的模型：键裂解、片段解离和 pro
《生物化学》38 (21) (1999) 6918–6934。网址
https://doi.org/10.1021/bi981966e

[42] Y. Ma、S. Gowda、R. Anantharaman、C. Laughman、V. Shah、C. Rack- auckas，《ModelingToolkit：一种用于基于方程的建模的可组合图变换系统》（2021年）。arXiv:2103.05244。[43] C. Rackauckas、Q. Nie，《DifferentialEquations.jl——用于在 Julia 中求解微分方程的高性能、功能丰富的生态系统》，《开放研究软件期刊》5 (1) (2017)。[44]M. Schlie mann、E. Bullinger、S. Borchers、F. Allgöwer、R. Findeisen、P. Scheurich，异质性降低了细胞死亡对 TNF 刺激的敏感性，《BMC系统生物学》5 (204) (2011)。网址 https://doi.org/10.1186/1752-0509-5-204

[45] J. K.Scott, P. I. Barton, 非线性控制系统可达集的界，《自动化》49 (1) (2013年) 93–100。网址：https://doi.org/10.1016/j.automatica.2012.09.020

[46] M. Tschaikowski, M.Tribastone, 异质非线性模型的近似约化（不同
《IEEE自动控制汇刊》61 (4) (2016) 1099–1104。网址：
https://doi.org/10.1109/TAC.2015.2457172

[47] A. Girard, G. J.Pappas, 近似双模拟：连接计算机科学与控制理论的桥梁
《欧洲控制期刊》17 (5) (2011) 568–578。网址：
https://doi.org/10.3166/ejc.17.568-578

[48]S. Soliman, F. Fages, O. Radulescu,一种通过热带平衡进行模型约化的约束求解方法，《分子生物学算法》9 (1) (2014)。网址：https://doi.org/10.1186/s13015-014-0024-2

[49] A. Goyer，用于线性微分算子符号-数值因式分解的 Sage 软件包，美国计算机协会《计算代数通讯》55 (2) (2021年) 44–48。网址 https://doi.org/10.1145/3493492.3493496

[50]F. Chyzak，A. Goyer，M. Mezzarobba，微分算子的符号-数值因式分解，载于：2022年国际符号与代数计算研讨会论文集，ISSAC '22，2022年，第73–82 页。网址 https://doi.org/10.1145/3476446.3535503