

扩散系数识别模型

摘要

扩散系数的参数识别问题是一类典型的反问题，高效且适定的扩散系数识别算法对解决此类问题具有重要意义。本文针对单组元一维扩散过程的不同测量工况，基于有限差分方法、数据归一化、曲线拟合及偏微分方程等理论，以及借鉴了泛函极值问题中的最速降线思想，通过确定扩散系数识别精度、稳定性和计算效率等指标，以准确识别扩散系数、提高计算效率、增加算法适定性为目标建立了识别模型，完成了扩散系数求解。

针对问题一：首先根据测量数据观察扩散通量 J 与组元浓度 C 空间分布的特征，并结合 Fick 第一定律获得扩散系数的测量值。使用 sigmoid 函数对组元浓度的空间分布进行曲线拟合，以修正后拟合曲线在空间上的导数代替组元浓度的数值梯度，进而结合扩散通量的测量数据获得扩散系数的计算值。通过精度评价指标发现模型计算得到的扩散系数均方误差达到 10^{-5} 量级，可以认为本文中使用的曲线拟合识别扩散系数的模型精度较高；另外，以解析梯度代替数值梯度可以有效增加扩散系数识别效率问题。

针对问题二：在仅有组元浓度或扩散通量二者之一数据的情况下，模型的核心在于将另外一个参数理论值构建出来，从而转化为类似于问题一的扩散通量 J 与组元浓度 C 均已知的情况。针对仅有组元浓度数据的情况，根据 Fick 第二定律，对组元浓度测量数据进行时间上数值差分、空间上迭代积分，构建出扩散通量的预测模型；针对仅有扩散通量数据的情况，对扩散通量的时间变化率在空间、时间上均建立了拟合函数模型，较好地预测了扩散通量不同时刻地发展情况，进而空间上数值差分、时间上迭代积分得到组元浓度理论值。将计算得到的扩散系数与问题一进行对比发现：仅有组元浓度数据情况下扩散系数的计算值偏高，而仅有扩散通量数据情况下扩散系数的计算值偏低，但扩散系数随组元浓度的变化趋势一致。

针对问题三：基于实验数据反演计算扩散系数时，测量数据的误差水平和数据量大小会对曲线拟合效果产生直接的影响。文中通过在原始数据基础上增加服从标准正态分布的随机误差，以不同的正态分布的峰值调整数据的误差水平；通过均匀采样和三次样条插值的方式在保留原有数据分布特征的前提下实现数据量的减少和增加。通过问题一的模型进行扩散系数的计算，与问题一中的扩散系数测量值对比发现，误差水平越大，拟合曲线的精度和稳定度都有所下降，导致扩散系数的计算精度整体上呈现降低的趋势；在一定的范围内，计算精度随着数据量的减小而提高，减小到一定值之后反而会导致精度的下降，而随着数据量的增加，计算精度会产生一定程度的下降。

针对问题四：对于多组分物系，由于组分之间存在交互作用，扩散系数的计算相当复杂。为了建立扩散系数识别模型，本文将适用于单组分与二组分物系的 Frick 扩散定律推广到多组分体系。将 Frick 第一定律扩展为张量形式，进而相应地改进本文针对单组元一维扩散过程的扩散系数识别模型，使用数值求解得到多组元高维扩散过程的扩散函数。

总体而言，本文提出的扩散系数反演计算模型通过巧妙地设计拟合函数，灵活运用最速降线思想，无需计算大量数值梯度，实现了单组元一维扩散方程扩散系数的识别问题，较好地适应了不同的测量工况，对解决当前扩散系数求解算法存储量大、计算效率不高的问题有一定的借鉴意义。

关键词: 扩散系数 参数识别 偏微分方程 曲线拟合 最速降线

目录

扩散系数识别模型.....	I
摘要.....	I
1 问题重述与分析.....	1
1.1 问题重述.....	1
1.2 问题分析.....	1
2 模型假设与符号说明.....	2
2.1 模型基本假设.....	2
2.2 符号说明.....	2
3 模型建立与求解.....	2
3.1 解题思路.....	2
3.1.1 扩散通量和组元浓度均已知的扩散系数识别问题.....	4
3.1.2 已知组元浓度的扩散系数识别问题.....	4
3.1.3 已知扩散通量的扩散系数识别问题.....	5
3.1.4 扩散系数识别精度分析问题.....	5
3.1.5 多组元高维扩散过程扩散系数问题.....	6
3.2 算法说明.....	6
3.2.1 有限差分法.....	6
3.2.2 数据归一化.....	7
3.2.3 牛顿迭代法.....	8
3.2.4 三次样条插值.....	9
3.3 模型建立.....	10
3.3.1 扩散通量和组元浓度均已知的扩散系数识别模型.....	10
3.3.2 已知组元浓度的扩散系数识别模型.....	13
3.3.3 已知扩散通量的扩散系数识别模型.....	14
3.3.4 扩散系数识别精度分析模型.....	16
3.3.5 多组元高维扩散过程扩散系数识别模型.....	17
3.4 模型效果指标定义.....	17
3.4.1 精度评价指标.....	17
3.4.2 稳定性指标.....	18
3.4.3 计算效率指标.....	18
4 模型应用及分析.....	18
4.1 已知扩散通量和组元浓度的扩散系数计算.....	18
4.2 已知组元浓度的扩散系数计算.....	24
4.3 已知扩散通量的扩散系数计算.....	27
4.4 扩散系数识别精度分析.....	31
4.4.1 误差水平变化的影响.....	31

4.4.2 数据量变化的影响	34
5 模型评价和改进	40
5.1 模型的优点	40
5.2 模型的不足	41
5.3 模型的改进和推广	41
参考文献	47
附 录	48
附录 A: 支撑材料列表	48
附录 B: 主要程序/关键代码	48

1 问题重述与分析

1.1 问题重述

在温度给定的条件下，扩散系数通常是关于组元的函数，一般情况下可采用实验测量间接得到^[1,2]。在实验测量中可得到与扩散系数相关联的物理量，如目标体系中组元的扩散通量、浓度分布等，再基于扩散系数与测量之间的关系通过计算可得到扩散系数。扩散系数的求解属于反问题中的参数识别问题^[3]。参数识别问题是一种将理论模型与试验数据结合起来用于预测的技术^[4]。目前解决此种反问题的求解有很多方法，如遗传算法^[4]、脉冲谱法^[2]等，但大多都存在稳定性不强，存储量大以及计算效率不高等问题。

分析题目和附件数据，将所得信息整理如下：

Fick 定律是描述物质扩散现象的宏观规律，它于 1855 年被生理学家 Fick 所发现并在实际问题中有着广泛的应用。设扩散系数为 D （单位： m^2/s ），扩散物质的体积浓度为 C （单位：原子数/ m^3 或 kg/m^3 ），**Fick 定律**的表述如下：

Fick 第一定律：

单位时间内通过垂直于扩散方向的单位截面积的扩散物质流量（扩散通量 J ）与该截面处的浓度梯度成正比，关系式如下：

$$J = -D\nabla C \quad (1)$$

Fick 第二定律：

给出了非稳态扩散过程中浓度随时间的变化率与扩散通量随距离变化率之间的关系为：在非稳态扩散过程中，在指定位置处，浓度随时间的变化率等于该处的扩散通量随距离变化率的负值，如下所示：

$$\frac{\partial C}{\partial t} = -\nabla \cdot J \quad (2)$$

附件数据：

【附件 1：扩散通量 J 】提供了 $T = 0, 0.2, 0.4, 0.6, 0.8$ 和 1 时刻扩散通量 J 的空间分布；

【附件 2：组元浓度 C 】提供了 $T = 0, 0.2, 0.4, 0.6, 0.8$ 和 1 时刻体积浓度 C 的空间分布。

核心问题：

本文的核心问题在于根据已有附件的全部或部分测量数据建立起扩散系数识别模型，使模型兼具预测准确和计算高效的特点，即：利用较短时间计算所得的扩散系数 D 预测值与真值之间的误差尽量小。

1.2 问题分析

扩散系数的识别问题涉及偏微分方程求解、曲线拟合等手段，往往由实验数据分析，模型建立，样本预测，模型优化这几个要素构成，本文需要从计算效率、识别精度和算法稳定性等角度出发，主要解决以下 4 个问题：

(1) 针对问题一，问题一的要求是建立扩散通量和浓度空间分布已知的情况下扩散系数识别的数学模型，针对目前算法存在的存储量大、稳定性不强和计算效率不高的问题设计合适的算法，通过题目提供的扩散通量和组元浓度测量数据完成对扩散系数

的求解，并观察扩散系数随浓度变化的曲线。

(2) 针对问题二，实际情况下希望测量数据尽量少而计算结果尽量好，存在两种情形。第一种是扩散通量未知而浓度空间分布已知，第二种是扩散通量已知而浓度空间分布未知。问题二的要求是分别对两种情形建立扩散系数识别的数学模型，巧妙利用模型的特征设计求解算法。利用组元浓度的测量数据完成第一种情形的求解，利用扩散通量的测量数据完成第二种情形的求解，最后比较分析与问题一求解结果的异同。

(3) 针对问题三，问题三的要求是在前两个问题的基础上总结出影响扩散系数识别精度的主要因素，需要从测量数据的误差水平、测量数据的数据量和迭代初值的选取方式进行具体分析。

(4) 针对问题四，问题一和问题二都是只考虑单组元、一维条件下物质扩散过程的扩散系数识别，而实际问题涉及到更加复杂的多组元、多维条件下扩散过程，要求思考并提出的模型和算法的拓展应用，并根据具体情况进行说明。

2 模型假设与符号说明

2.1 模型基本假设

本文拟解决物质扩散现象的扩散系数识别问题，所建立的模型满足以下几点基本假设：

(1) 整个过程仅考虑给定温度下扩散效应对物质运输的影响；

(2) 所涉及的扩散过程属于无源扩散，随着时间的推移，所有位置的组元浓度将会稳定在某个值附近，即：当时间 $T \rightarrow +\infty$ 时， $C(x) = \text{Const}$ ；

(3) 与扩散系数 D 相关联的物理量在题目给定的参数（组元浓度 C 、扩散通量 J 、位置 x 、时间 t ）中选取；

2.2 符号说明

本文定义了如下 6 个使用次数较多的符号，其余符号在使用时注明。

表1 部分符号说明

符号	含义	单位
T	时刻	s
t	时间	s
x	位置	m
C	组元浓度	原子数/m ³ 或 kg/m ³
J	扩散通量	原子数/m ² 或 kg/m ²
D	扩散系数	m ² /s

3 模型建立与求解

3.1 解题思路

首先对附件数据进行分析。从两个附件中扩散通量 J 和组元浓度 C 的测量数据可知这是一个单组元、一维的条件下的扩散系数识别问题，前三个问题均需要使用该数据建立模型计算结果。

根据题目提供的附件，扩散通量和组元浓度在空间上的分布如图 1 所示。

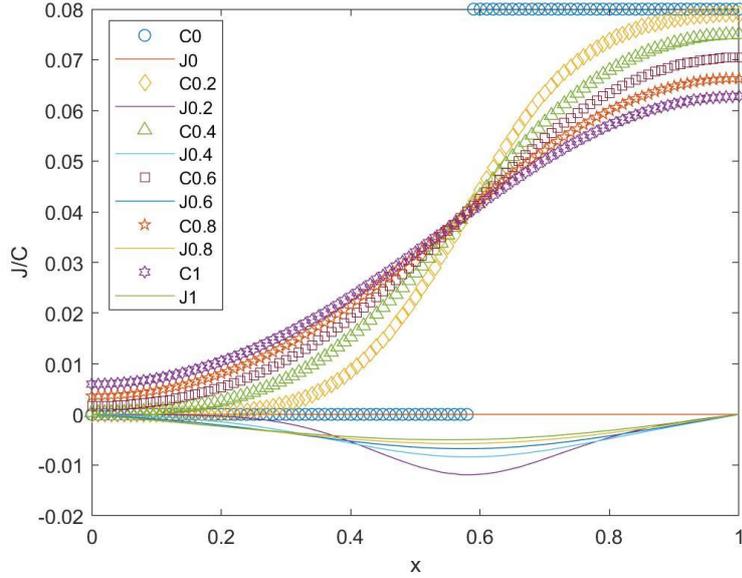


图1 不同测量时刻扩散通量和组元浓度的空间分布

上图中图例中符号“C”代表组元浓度 C 的数据，符号“J”代表扩散通量 J 的数据，数字表示不同时刻，例如“J0.2”代表是 $T=0.2$ 时刻的扩散通量 J 的数据。从图中可得到扩散通量 J 和组元浓度 C 的初始条件为：

$$J|_{t=0} = 0, \quad C|_{t=0} = \begin{cases} 0 & (0 < x \leq 0.58) \\ 0.08 & (0.58 < x \leq 1) \end{cases} \quad (3)$$

根据上图和附件 1 中扩散通量 J 的数据留意到扩散通量 J 也有隐含的边界条件：

$$J|_{x=0} = 0, \quad J|_{x=1} = 0 \quad (4)$$

这是因为在封闭的一维空间范围内，物质扩散过程在空间上是遵循扩散组元由高浓度区向低浓度区扩散，这也是 Fick 第一定律中负号代表扩散方向为浓度梯度反方向的原因。而在本题边界 $x=0$ 和 $x=1$ 处是不存在浓度梯度的，即：

$$\nabla C|_{x=0} = \vec{0}, \quad \nabla C|_{x=1} = \vec{0} \quad (5)$$

因为一维条件下，向量即可当作标量， $\vec{0}=0$ ，因此将 (5) 代入公式 (1) 即可推导出边界条件 (4)。

则 Fick 第一定律的公式 (1) 可以简化为：

$$J = -D\nabla C = -D \frac{\partial C}{\partial x} \quad (6)$$

Fick 第二定律的公式 (2) 可以简化为：

$$\frac{\partial C}{\partial t} = -\nabla \cdot J = -\frac{\partial J}{\partial x} \quad (7)$$

假设组元浓度 C 和扩散通量 J 均为关于位置 x 和时间 t 的函数，即 $C = C(x, t)$ 、 $J = J(x, t)$ ，由此可抽象出以下初边值条件：

$$\begin{cases} \frac{\partial C(x,t)}{\partial t} - \frac{\partial(D \frac{\partial C(x,t)}{\partial x})}{\partial x} = 0, & 0 \leq x \leq 1, 0 \leq t \leq 1 \\ C(x,0) = \varphi(x), & 0 \leq x \leq 1 \\ C(0,t) = g_0(t), & 0 \leq t \leq 1 \\ C(1,t) = g_1(t), & 0 \leq t \leq 1 \end{cases} \quad (8)$$

则本文所需解决扩散系数识别问题的核心在于求解在已知 (8) 条件下的单组元、一维扩散方程。

3.1.1 扩散通量和组元浓度均已知的扩散系数识别问题

通过附件所得的扩散通量 J 和组元浓度 C 的测量数据不仅可以直接获得在不同测量时刻扩散通量和组元浓度在空间上的分布, 同时还可通过数值差分获悉在不同时刻二者的空间梯度 $\frac{\partial C}{\partial x} \Big|_{t=t_i}$ 和 $\frac{\partial J}{\partial x} \Big|_{t=t_i}$ 。

根据 Fick 第一定律, 结合组元浓度 C 数值差分所得的空间梯度大小以及扩散通量 J 的数据值可计算得到扩散系数 D 。既然组元浓度 C 以及扩散通量 J 在不同时刻的空间分布均已知且附件中的测量数据在本文中假定无误, 由此推定, 计算出扩散系数的测量值也是根据上述二者的测量数据所得, 因此该扩散系数可合理地作为问题一中扩散系数 D 的真值。

目前大多扩散系数反演问题通过极小化目标函数的迭代算法进行求解, 需要计算大量的数值梯度, 导致存储数据量大、计算效率不高^[4]。本文针对上述问题, 以减少数值梯度的计算量为核心, 考虑从组元浓度 C 的空间分布特征入手, 根据实验数据中不同时刻的组元浓度 C 的空间分布分别进行曲线拟合, 得到不同时刻组元浓度 C 关于空间位置 x 的关系式, 并利用该关系式的空间导数代替数值差分的浓度梯度, 进而可以得到较为连续的扩散系数 D 的模型计算值。这种基于曲线拟合的扩散系数计算方法可以大幅减少数值梯度的计算, 从而降低需要存储数据的数量, 提高解决扩散系数识别问题的效率。

综上, 问题一采用曲线拟合法得到组元浓度 C 在空间上的分布函数, 该分布函数对 x 求导并将其作为组元浓度 C 的空间上的偏微分用于计算扩散系数的理论值, 当理论值与根据实验测量数据的所得扩散系数 D 的整体贴合程度较高时, 即认为问题一得到了解决。

3.1.2 已知组元浓度的扩散系数识别问题

当组元浓度的空间分布已知而扩散通量未知时, 此时的难点在于仅通过 Fick 第一定律从无法得到扩散系数 D 的理论计算结果, 则需考虑通过 Fick 第二定律获取扩散通量 J 的空间散度与组元浓度 C 的时间变化率之间的关系。此处可以用到时间差分获取 $\frac{\partial C}{\partial t} \Big|_{x=x_i}$, 则 $\frac{\partial J}{\partial x} \Big|_{t=t_i}$ 也相应可得, 进而按照时间步长积分可得相应的扩散通量 J 的理论数据。因此, 根据组元浓度 C 的测量数据时间上数值差分、空间上迭代积分, 构建出扩散通量 J 的预测模型。

解决此问题的核心在于, 将仅有组元浓度 C 的情况转换成组元浓度 C 和扩散通量 J 均已知的情况, 就可以通过问题一所抽象出的扩散通量和组元浓度均已知的扩散系数识别问题的求解算法进行反演计算。在单组元、一维的条件下, 由 Fick 第二定律得到组元浓度 C 在时间上的偏微分与扩散通量 J 在空间上的偏微分成相反数的关系。由于

组元浓度 C 已知，扩散通量 J 可通过组元浓度随时间的变化率并在空间上的积分后得到。同样采用问题一中的曲线拟合算法进行对组元浓度 C 的处理，将扩散通量 J 的理论计算值与之相结合求解就能获得仅有组元浓度 C 测量数据的情况下的扩散系数 D 的模型计算值，将其与问题一的扩散系数的理论计算值进行比较，分析结果的差异以及产生差异的原因，即可完成问题二中第（1）问的求解。

3.1.3 已知扩散通量的扩散系数识别问题

当扩散通量 J 的空间分布已知而组元浓度 C 未知时，仅依靠 Fick 第二定律虽然可以获得扩散通量 J 空间上的散度与组元浓度 C 时间上的变化率之间的关系，类似于上一小节 3.1.2 的做法，将扩散通量 J 空间上的测量数据数值差分代替偏微分 $\frac{\partial J}{\partial x}\Big|_{t=t_i}$ ，但在时间上迭代积分却不能获得完全的组元浓度 C 理论计算值。这是因为仅有组元浓度 C 的情况和仅有扩散通量 J 的情况所不同的是，前者在将组元浓度 C 的测量数据时间上数值差分、空间上迭代积分的时候，是能通过物质扩散现象的物理性质得到扩散通量 J 的空间位置 $x=0$ 的初值条件是 $J\Big|_{x=0} \equiv 0$ ；但后者不然，在将扩散通量 J 的测量数据空间上数值差分、时间上迭代积分的时候，组元浓度 C 在时间上 $T=0$ 时刻的初值却是未知的而且不能从物理过程中直接获得，因此需要从另外的角度将组元浓度 C 在时间上 $T=0$ 时刻的初值进行确定。

根据模型假设，当经过一段足够长的时间后，所有位置的组元浓度 C 都将稳定在一个水平，那么可以进行反向推演：通过对 $\frac{\partial J}{\partial t}$ 建立数学模型实现合理地预测扩散通量 J 的变化规律，进而推演出后续不同 T 时刻的 $\frac{\partial J}{\partial x}\Big|_{t=t_i}$ ，利用 Fick 第二定律将 $\frac{\partial C}{\partial t}\Big|_{x=x_i}$ 相应可以获得，进而通过时间积分获得组元浓度 C 在不同 T 时刻的变化量，当时间足够长时，每一个位置的组元浓度 C 的变化量将会分别达到一个稳定的状态。理论上这就意味着这是每个位置的组元浓度 C 初值到达稳定水平的变化量，因此当确定一个合适的稳定水平的大小后，组元浓度 C 的初值就能反算得到。此时便又能转换成组元浓度 C 和扩散通量 J 均已知的情况，就可以再次通过问题一的扩散系数识别模型求解，将其与问题一的扩散系数 D 的理论计算值进行比较，分析二者结果的差异及根源，即可完成问题二中第（2）问的求解。

3.1.4 扩散系数识别精度分析问题

本文中，扩散系数通过实验数据反演计算，计算过程中除了模型误差外，主要影响扩散系数识别精度的因素包括随机误差、数据量以及迭代过程中迭代初值的选取。

随机误差反映了测量数据的误差水平，本文考虑通过在已有测量数据的基础上引入给定分布的随机误差，然后重复问题一的求解过程得到加入随机误差后的扩散系数。将考虑不同水平的随机误差计算所得扩散系数与不考虑随机误差计算所得扩散系数进行对比，采用合适的精度评价指标分析测量数据误差水平对扩散系数计算精度的影响。

测量数据的数据量一般会影响数据拟合的程度，应用同样的拟合方法对不同大小的样本进行曲线拟合时，可能会出现过拟合、欠拟合之类的问题，本文在计算扩散系数时充分利用了曲线拟合的思想，所以考虑数据量对扩散系数计算精度的影响是有必要的。文中通过对原始数据设计采样方式和插值算法，在尽可能保持原始数据分布特征（意味着误差水平保持一致，控制变量法的思想）的情况下得到不同大小的样本，采用合适的精度评价指标分析数据量对扩散系数计算精度的影响。

关于迭代初值选取对扩散系数计算精度的影响，大多文献从极小化目标函数的优化策略出发，采用不同的优化算法进行迭代计算，在此过程中会涉及扩散系数迭代初值的选取。但是也有学者提出了一些不依赖于迭代初值的扩散系数反演算法，比如周焕林等提出了一种对迭代初值不敏感的改进布谷鸟算法、刘艳杰等提出了一种不依赖初值选取的贝叶斯反演。本文中，核心思路是通过曲线拟合和数值方法的结合进行扩散系数的计算，不涉及以扩散系数为设计变量的优化算法，故扩散系数的初值选取对本文的扩散系数计算精度影响不大。但值得一提的是，由于在问题二中用到了迭代算法去计算扩散通量随时间的变化率与扩散通量的关系，涉及到对形参 θ 初值的选取。但是由于不是扩散系数的迭代初值对迭代算法收敛性的影响，不在本文研究范围内，故可以认为本文对迭代初值的选取不敏感。

3.1.5 多组元高维扩散过程扩散系数问题

实际应用中，如材料计算科学的传质、吸收、催化等反应的模拟中，扩散问题求解大多基于高维和多组元等复杂情形。对于多组元物系，由于组分之间存在交互作用，扩散系数的求解相比单组元要更加复杂。对于此类问题的求解，应先从 Fick 扩散定律入手。适用于单组元的 Fick 扩散定律此时已经不能解决当前问题，应将其推广到多组元物系。

已有部分研究对多组元物系传质过程中的扩散系数求解问题进行研究。已有学者讨论了多组元物系扩散传质过程中描述多组元物系传质过程的基本方程 MS 方程和 Fick 扩散系数的推广。还有学者提出利用 Maxwell—Stefan 扩散系数近似代替 Fick 扩散系数，因为 Fick 扩散系数会受到热力学非理想性和分子间相互作用力两种因素的影响，其计算与研究较不方便。

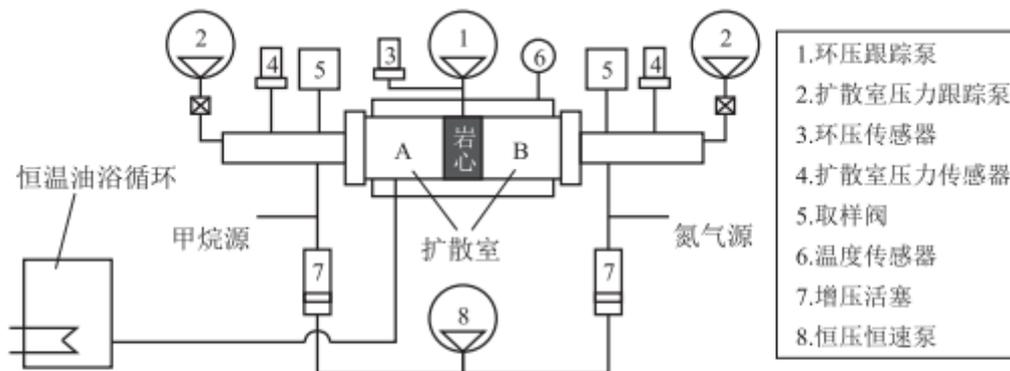


图3 扩散系数模拟实验装置简易图
Fig.3 Simplified diagram of simulator for diffusion coefficient analysis

图2 多组元高维的扩散问题应用实例——纳米孔隙中页岩气扩散模拟实验^[6]

3.2 算法说明

本文为求解问题而建立模型时涉及的计算方法主要有差分法、数据归一化、三次样条插值和牛顿迭代法，下文将逐一进行简要说明。

3.2.1 有限差分法

有限差分法 (Finite Difference Method) 常用于数值计算方法中求偏微分 (或常微分) 方程和方程组定解问题的数值解，是微分以及偏微分导数的一种离散化方法，即

用相邻两个或者多个数值点的差分取代偏微分方程中导数或者偏导数的一种算法。有限差分方法具有简单、灵活以及通用性强等特点，容易在计算机上实现。在实际数值求解偏微分方程的过程中，选择差分方法是离散化偏微分方程的第一步。其中最常见的差分方法有前向差分、后向差分、中心差分，如图3所示。

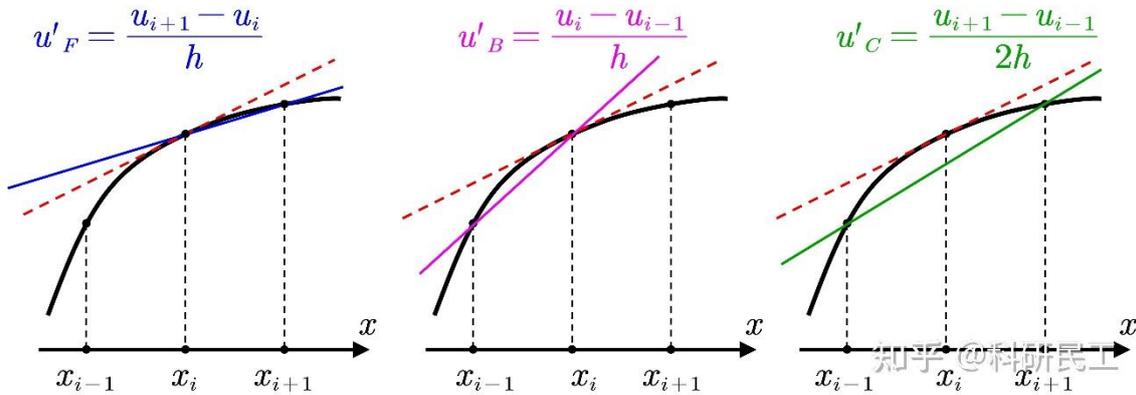


图3 三种不同差分方法的示意图^[7]

上图中， u 和 x 分别表示因变量和自变量， h 为自变量的步长 $h = \Delta x$ 。理论上讲，当步长取值足够小时，差分计算就可以取代导数或偏导数计算。

$$u'(x) = \lim_{\Delta x \rightarrow 0} \frac{\Delta u}{\Delta x} \quad (9)$$

前向差分（前差，FDM，Forward Differential Method）：

$$\text{FDM: } \frac{\partial u}{\partial x} = \frac{u_{i+1} - u_i}{h} \quad (10)$$

后向差分（后差，BDM，Back Differential Method）：

$$\text{BDM: } \frac{\partial u}{\partial x} = \frac{u_{i-1} - u_i}{-h} \quad (11)$$

中心差分（中心差，CDM，Central Differential Method）：

$$\text{CDM: } \frac{\partial u}{\partial x} = \frac{u_{i+1} - u_{i-1}}{2h} \quad (12)$$

上述公式 (10)(11)(12) 均为一阶精度的差分格式，实际求解问题还有更高阶的差分方法，在本文不再赘述。

3.2.2 数据归一化

数据归一化（也称标准化，Normalize）是将数据按比例缩放，使之落入一个特定的区间中，通常为 $[0, 1]$ 。与之相伴随的还有无量纲化（Nondimensionalize 或 Dimensionless）去除数据的单位限制，使其转化为纯数值，进而对另外一些数据进行比较和评价具有合理性，便于不同单位或者量级的指标进行对比或加权。

针对一维数据的求和归一化的计算方法类似于计算权重，其公式如下：

$$\text{normal}(X) = \frac{X}{\text{sum}(X)} \quad (13)$$

针对一维数据的极值归一化（Min-Max 归一化）的计算式如下：

$$\text{normal}(X) = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (14)$$

还有最常用的 Z-Score 归一化，如图 4 所示（图中 $J(w,b)$ 为需要归一化的目标函数），公式为：

$$\text{normal}(X) = \frac{X - \mu}{\sigma} \quad (15)$$

Why normalize inputs?

$$J(w, b) = \frac{1}{m} \sum_{i=1}^m \mathcal{L}(\hat{y}^{(i)}, y^{(i)})$$

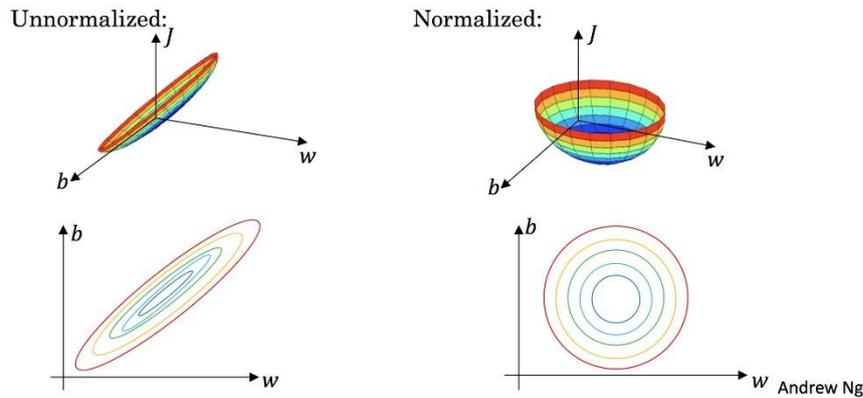


图4 Z-Score 归一化效果示例^[8]

还有更多较为复杂的归一化方法例如对二维图像数据处理的归一化方法，用到的公式为：

$$R(i, j) = \frac{\sum_{m=1}^M \sum_{n=1}^N S^{ij}(m, n) T(m, n)}{\sqrt{\sum_{m=1}^M \sum_{n=1}^N S^{ij}(m, n)} \sqrt{\sum_{m=1}^M \sum_{n=1}^N [T(m, n)]^2}} \quad (16)$$

其中 M 和 N 为图片大小，当 $R(i, j)$ 均越接近于 1 时图像处理前后相似性越强。

3.2.3 牛顿迭代法

牛顿迭代法（Newton's Method）是牛顿在 17 世纪提出的一种在实数域和复数域上近似求解方程的方法，又称为牛顿-拉弗森方法（Newton-Raphson Method）。其最大优点是在方程的单根附近具有平方收敛，该方法被广泛用于计算机编程中。

牛顿迭代算法如图 5 示意，具体步骤如下所示：

Step 1 设 x^* 是 $f(x) = 0$ 的根，选取 x_0 作为 x^* 初始近似值，并设 $f(x)$, $f'(x)$ 和 $f''(x)$ 在 x^* 附近连续。过点 $(x_0, f(x_0))$ 做曲线 $y = f(x)$ 的切线 L ， L 的方程为 $y = f(x_0) + f'(x_0)(x - x_0)$ ，求出 L 与 x 轴交点的横坐标 $x_1 = x_0 - f(x_0) / f'(x_0)$ ，称 x_1 为 x_0 的一次近似值。

Step 2 过点 $(x_1, f(x_1))$ 做曲线的切线，并求该切线与 x 轴交点的横坐标 $x_2 = x_1 - f(x_1) / f'(x_1)$ ，称 x_2 为 x^* 的二次近似值。

Step 3 重复以上过程，得 x^* 的近似值序列 $x_{k+1} = x_k - f(x_k) / f'(x_k)$ ，其中 x_k 称为 x^* 的 $k+1$ 次近似值。

Step 4 在迭代序列收敛的情况下，取满足精度的迭代值 x_k 作为方程的根 x^* 的近似值。

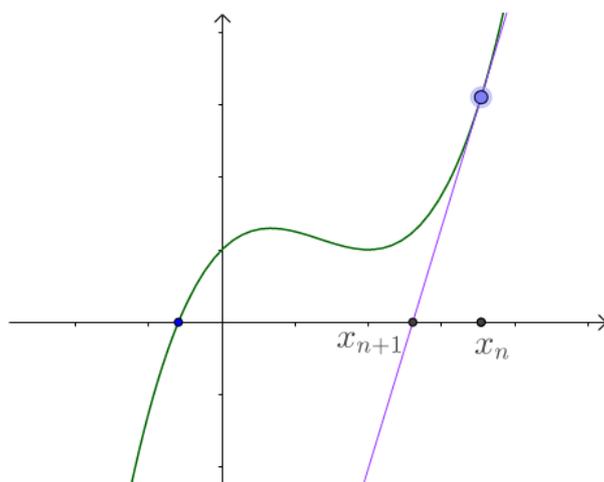


图5 牛顿迭代法示意图^[9]

3.2.4 三次样条插值

三次样条插值 (Cubic Spline Interpolation) 是通过一系列数据点的一条光滑曲线，数学上通过求解三弯矩方程组得出曲线函数组的过程。实际计算时需要引入边界条件才能完成计算，数值计算软件如 MATLAB 把非扭结边界条件作为默认的边界条件。例如对 $y = \sin(x)$ 进行三次样条插值的效果如下图 6 所示，图中的数据点就是通过三次样条插值算法在函数上所得若干数据点。

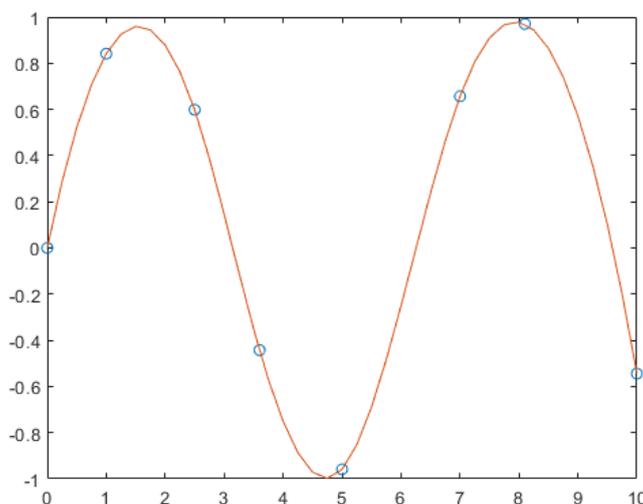


图6 三次样条插值效果示例^[10]

三次样条插值算法步骤如下：

假定有 $n+1$ 个数据节点： $(x_0, y_0), (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$

Step 1 计算步长： $h_i = x_{i+1} - x_i$

Step 2 将数据节点和指定的首位端点条件代入矩阵方程

Step 3 求解矩阵方程，求得二次微分值 m_i 。该矩阵常见解法为高斯消元法，可以对系数矩阵进行 LU 分解，分解为单位下三角矩阵和上三角矩阵：

$$B = Ax = (LU)X = L(Ux) = Ly \quad (17)$$

Step 4 计算样条曲线的系数：

$$\begin{cases} a_i = y_i \\ b_i = \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{2} m_i - \frac{h_i}{6} (m_{i+1} - m_i) \\ c_i = \frac{m_i}{2} \\ d_i = \frac{m_{i+1} - m_i}{6h_i} \end{cases} \quad (18)$$

Step 5 在每个子区间: 在 $x_i \leq x \leq x_{i+1}$ 中创建方程:

$$h_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad (19)$$

3.3 模型建立

根据 3.1 节的解题思路, 为解决不同情况下的扩散系数识别问题, 下面介绍本文建立相应数学模型的详细过程。

3.3.1 扩散通量和组元浓度均已知的扩散系数识别模型

假设组元浓度 C 和扩散通量 J 均为关于位置 x 和时间 t 的函数, 即 $C = C(x, t)$ 、 $J = J(x, t)$, 从 3.1 图 1 和附件中可得到以下信息:

(1) 初始时刻, 组元浓度 C 在空间上的分布为

$$C(x, 0) = \begin{cases} 0, & 0 \leq x \leq 0.58 \\ 0.08, & 0.58 < x \leq 1 \end{cases} \quad (20)$$

(2) 组元浓度 C 的边界条件分别为

$$\begin{cases} C(0, t) = g_0(t), & 0 \leq t \leq 1 \\ C(1, t) = g_1(t), & 0 \leq t \leq 1 \end{cases} \quad (21)$$

其中 $g_0(t)$ 和 $g_1(t)$ 分别为组元浓度 C 在两端边界处随时间的变化函数。

(3) 初始时刻, 组元浓度 C 在空间上的分布为

$$J(x, 0) = 0, \quad 0 \leq x \leq 1 \quad (22)$$

(4) 扩散通量 J 的边界条件分别为

$$\begin{cases} J(0, t) = 0 & 0 \leq t \leq 1 \\ J(1, t) = 0, & 0 \leq t \leq 1 \end{cases} \quad (23)$$

此外, 从 3.1 图 1 还观察到, 组元浓度 C 关于位置 x 的曲线随时间变化有逐渐逼近一条水平线的趋势, 而扩散通量 J 在空间分布上最终都会逼近 0, 这也与扩散过程的实际现象相对应。

根据 Fick 第一定律的简化公式 (6), 在扩散通量 J 和组元浓度 C 的空间梯度已知的情况下扩散系数 D 可表示为:

$$D = -J / \left(\frac{\partial C}{\partial x} \right) \quad (24)$$

3.3.1.1 扩散系数测量值计算

式 (24) 中的组元浓度 $C(x, t)$ 关于位置 x 的偏导数可通过前差公式 (10) 和中心差公

式 (12) 计算, 使用下列公式表示:

$$\begin{aligned} \text{FDM: } \frac{\partial C_i}{\partial x} &= \frac{C_{i+1} - C_i}{\Delta x} \\ \text{CDM: } \frac{\partial C_i}{\partial x} &= \frac{C_{i+1} - C_{i-1}}{2\Delta x} \end{aligned} \quad (25)$$

式中的 C_i 表示空间上第 i 个位置点的组元浓度 C 值。根据附件所给出的数据, 本文取距离步长为 $\Delta x = 0.01$ 。

则 $C(x, t)$ 函数关于位置 x 的差分公式可近似作为组元浓度 C 的空间梯度。根据式 (24)(25) 以及题目提供的扩散通量 J 测量数据可以得到扩散系数 D 的测量值。

3.3.1.2 组元浓度空间分布拟合

为了减少数值梯度的计算量, 从组元浓度 C 的空间分布特征入手, 根据 3.1 图 1 观察和实验数据表明, 除了初始 $T = 0$ 以外的不同时刻 $T = 0.2、0.4、0.6、0.8、1$ 的组元浓度 C 的空间发展态势相似, 因此适合进行公式拟合。

常见的曲线拟合方式有多项式拟合、指数拟合、三角 (反三角) 函数拟合、特殊函数拟合等, 需要结合数据的特征设计合适的拟合函数。

根据浓度在空间上的分布 $C-x$ 曲线呈现中心对称的“S”形特征, 考虑反正切函数 $y = \arctan(x)$, 双曲正切 $y = \tanh(x)$, 如图 7 所示, 还有一类特殊的函数 sigmoid 函数 $y = \frac{1}{1 + e^{-x}}$ 曲线特征则可以较好地贴合这一特征。故本文考虑反三角函数、双曲正切函数和 sigmoid 函数进行拟合, 将拟合结果进行对比, 选取合适的拟合函数。

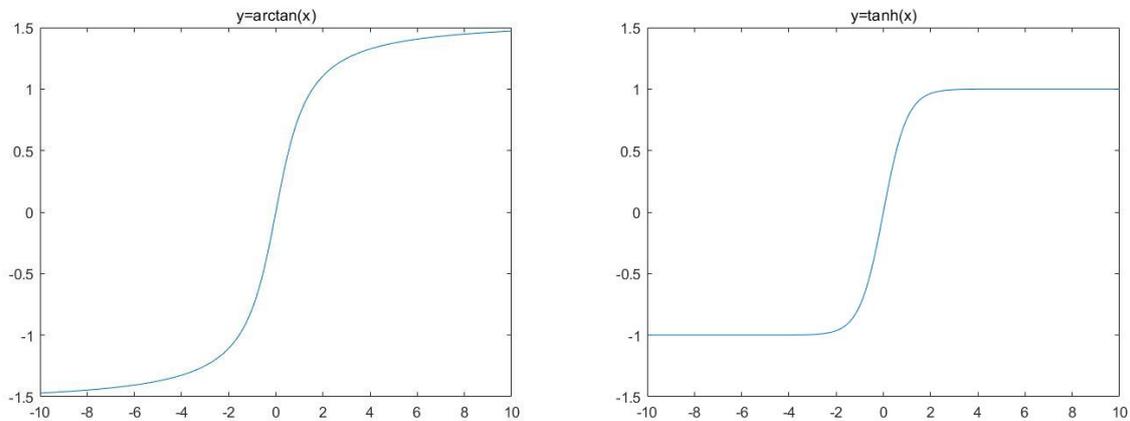


图7 “S”形特征拟合函数

将三种拟合函数的含参形式列出如下:

$$\begin{cases} y = a_1 \arctan [a_{2,1} (x - a_{2,2})] + a_3 \\ y = a_1 \tanh [a_{2,1} (x - a_{2,2})] + a_3 \\ y = \frac{a_1}{1 + e^{-a_{2,1}(x - a_{2,2})}} + a_3 \end{cases} \quad (26)$$

利用已知点将式 (26) 中的三种函数的待定参数计算出来, 即可得到相应的拟合函

数表达式。设组元浓度关于空间位置的拟合函数为

$$C(x, t_i) = \phi_i(x) \quad (27)$$

式中 $\phi_i(x)$ 即表示 t_i 时刻的浓度随空间的变化关系式，则有：

$$\frac{\partial C(x, t_i)}{\partial x} = \phi_i'(x) \quad (28)$$

显然， $\phi_i'(x)$ 相比于组元浓度 C 的空间梯度更容易获得，易知：

$$\begin{aligned} [\arctan(x)]' &= \frac{1}{1+x^2} \\ [\tanh(x)]' &= \text{sech}^2(x) \\ \left[\frac{1}{1+e^{-x}} \right]' &= \frac{e^{-x}}{(1+e^{-x})^2} \end{aligned} \quad (29)$$

由此关系式代替不同 T 时刻组元浓度 C 关于空间位置 x 的偏导数，并利用该关系式的空间导数代替数值差分的浓度梯度，进而可以得到较为连续的扩散系数 D 的模型计算值。

最后，根据式 (24) 即可得到扩散系数的计算值，即：

$$D = -\frac{J}{\phi_i'(x)} \quad (30)$$

3.3.1.3 组元浓度数据归一化

在绘制随组元浓度 C 变化的扩散系数 D 的曲线时，初始 $T=0$ 时刻由于组元浓度 C 要么为 0 要么为 0.08，数据点呈现两极分化没有参考意义，因此不纳入考量范畴；另外考虑到不同 T 时刻的组元浓度 C 不像位置 x 那样具有相同的区间（例如 $T=0.2$ 时刻，组元浓度 C 的变化区间约为 $[0, 0.08]$ ，但在 $T=0.8$ 时刻，组元浓度 C 的变化区间约为 $[0.036, 0.066]$ ，因此导致横轴数据不统一），为了方便比较不同 T 时刻的 $D-C$ 曲线的异同，并使之呈现趋势更明显，故本文对组元浓度 C 进行归一化处理。

采用数据极值的归一化公式 (14)，则有：

$$\text{normal}(C) = \frac{C - \min(C)}{\max(C) - \min(C)} \quad (31)$$

3.3.1.4 扩散系数-组元浓度曲线修正

在扩散系数的反演问题中，常常会引入正则项来克服计算过程中不适定性带来的困难。类似的，本文在计算系数的过程中面临拟合函数误差带来的稳定性问题，故考虑引入修正项对模型进行修正。

在式 (24) 通过拟合函数的确定而简化为式 (30) 的过程中，有一种情况将显著导致模型预测值的不准确，那就是当式 (24) 的扩散通量 J 和组元浓度 C 的空间梯度均为 0 时，此时的扩散系数 D 可能出现 NaN（无意义）或 $\pm \text{Inf}$ （无穷大）的情况，而根据高等数学可知实际上扩散系数 D 还可能是实数值并可通过洛必达法则等方式将其大小计算出来，因为回归 Fick 第一定律 (1) 可知，等号左右两端均为 0，等式恒成立。在本题

中，出现这种情况的位置恰好位于单组元、一维扩散方程的边界—— $x = 0$ 、 $x = 1$ 处，扩散通量 J 和组元浓度 C 的空间梯度均为 0，该现象在 3.1 节已进行分析。

而式 (30) 简化后带来的问题便是将上述现象抹除，通过 (29) 可知，使用的拟合函数求导结果在 $x \rightarrow \pm\infty$ 只能趋近于 0，也就是式 (30) 的分母项在边界附近不能很好地替代原函数，因此需要在边界附近进行修正。

可想而知，通过式 (30) 得出的扩散系数-组元浓度 D - C 曲线与实际情况相比，由于在边界处分子项逐渐趋近于 0 而分母项不等于 0，故最终会在边界附近会有骤降趋势，而实际情况是扩散系数 D 将会是连续变化的，故本文通过在边界附近将拟合曲线用延长线代替来进行修正，具体公式为：

$$D = \begin{cases} \frac{\partial}{\partial C} \left(\frac{J}{\phi_i'(x_m)} \right) (C - C_m) - \frac{J}{\phi_i'(x_m)}, & x \in [0, x_m) \\ -\frac{J}{\phi_i'(x)}, & x \in [x_m, x_n] \\ \frac{\partial}{\partial C} \left(\frac{J}{\phi_i'(x_n)} \right) (C - C_n) - \frac{J}{\phi_i'(x_n)}, & x \in (x_n, 1] \end{cases} \quad (32)$$

可认为是对 $x \in [0, x_m) \cup (x_n, 1]$ 区间的扩散系数-组元浓度 D - C 曲线进行修正，此处满足 $0 < x_m < x_n < 1$ 。

3.3.2 已知组元浓度的扩散系数识别模型

根据 Fick 第二定律的简化公式 (7)，在扩散通量 J 未知而组元浓度 C 的时间变化率已知的情况下，扩散通量 J 可表示为：

$$J = J|_{x=0} + \int_0^x \left(\frac{\partial J}{\partial x} \right) dx = J|_{x=0} + \int_0^x \left(-\frac{\partial C}{\partial t} \right) dx \quad (33)$$

根据边界条件 (23)，任一 T 时刻均有 $J|_{x=0} = 0$ ，因此使用时间差分获取 $\frac{\partial C}{\partial t}|_{x=x_i}$ ，则 $\frac{\partial J}{\partial x}|_{t=t_i}$ 也相应可得，进而按照时间步长积分可得相应的扩散通量 J 的理论数据。

式 (33) 中的组元浓度 $C(x, t)$ 关于时间 t 的偏导数可通过后差公式 (11) 计算如下：

$$\text{BDM: } \frac{\partial C_{i,j}}{\partial t} = \frac{C_{i,j} - C_{i,j-1}}{\Delta t} \quad (34)$$

此处的 $C_{i,j}$ 代表在空间位置上第 i 个点的第 j 个时刻的组元浓度 C 的值，本文的时间步长，根据附件数据取 $\Delta t = 0.2$ 。

根据差分的思想同理可得扩散通量 J 在空间上迭代积分的计算式：

$$J_{i,j} = \sum_1^i \left(\frac{\partial J_{i,j}}{\partial x} \right) \Delta x = \sum_1^i \left(-\frac{\partial C_{i,j}}{\partial t} \right) \Delta x \quad (35)$$

空间步长 $\Delta x = 0.01$ 在 3.3.1 已有说明，则 $C(x, t)$ 函数关于时间 t 的差分公式可近似作为组元浓度 C 的时间变化率。结合式 (33)(34)(35) 以及附件 2 提供的组元浓度 C 的测量数据，可以所得扩散通量 J 的理论值。

将上述求得的扩散通量 J 的理论值和组元浓度 C 的测量值作为数据基础，此时便可以使用上一节 3.3.1 所采用的扩散通量和组元浓度均已知的扩散系数识别模型进行扩散系数 D 的理论值计算，计算方法详见 3.3.1。

3.3.3 已知扩散通量的扩散系数识别模型

采用类似于 3.3.2 的计算步骤，根据 Fick 第二定律的简化公式 (7)，在组元浓度 C 未知而扩散通量 J 已知的情况下，组元浓度 C 可表示为：

$$C = C|_{t=0} + \int_0^t \left(\frac{\partial C}{\partial t} \right) dt = C|_{t=0} + \int_0^t \left(-\frac{\partial J}{\partial x} \right) dt \quad (36)$$

式 (36) 中的扩散通量 $J(x,t)$ 关于时间 t 的偏导数也可通过前差公式 (10) 和中心差公式 (12) 计算，使用下列公式表示：

$$\begin{aligned} \text{FDM: } \frac{\partial J_{i,j}}{\partial x} &= \frac{J_{i+1,j} - J_{i,j}}{\Delta x} \\ \text{CDM: } \frac{\partial J_{i,j}}{\partial x} &= \frac{J_{i+1,j} - J_{i-1,j}}{2\Delta x} \end{aligned} \quad (37)$$

式中的 $J_{i,j}$ 代表在空间位置上第 i 个点的第 j 个时刻的扩散通量 J 的值，空间步长 $\Delta x = 0.01$ 在 3.3.1 已有说明。

根据差分的思想同理可得组元浓度 C 在时间上迭代积分的计算式（时间步长 $\Delta t = 0.2$ 在 3.3.2 已有说明）：

$$C_{i,j} = \sum_1^j \left(\frac{\partial C_{i,j}}{\partial t} \right) \Delta t = \sum_1^j \left(-\frac{\partial J_{i,j}}{\partial x} \right) \Delta t \quad (38)$$

结合公式 (36)(37)(38) 可以发现，尚有一个量还处于未知情况——组元浓度 C 在时间上 $T = 0$ 时刻的初值 $C|_{t=0}$ 。在 3.1.3 节中已有分析，当采用上述计算方式的时候，在将扩散通量 J 的测量数据空间上数值差分、时间上迭代积分中，组元浓度 C 在时间上的初值却是未知的而且不能从物理过程中直接获得，此时获取的只是因此任一位置在经过一定时间后组元浓度 C 的变化量而非其当前大小，需要从另外的角度将组元浓度 C 在时间上 $T = 0$ 时刻的初值进行确定。

对于本文所提到的在给定温度下的扩散过程，当系统空间上存在浓度梯度 $\frac{\partial C}{\partial x} \neq 0$ 时，扩散通量 J 不为 0。而在无源假设下，本系统将会以有限的时间内逐步接近平衡状态，即扩散通量 J （或浓度梯度 $\frac{\partial C}{\partial x}$ ）将逐步接近 0，使得系统内的扩散运动由快到慢。且当达到足够长的 T 时刻，可以认为系统内部所有空间分布点处的扩散通量 J （或浓度梯度 $\frac{\partial C}{\partial x}$ ）将同时到达 0。结合 Fick 定律，可以将扩散过程中扩散通量 J 的特点总结为：

- (1) 空间中所有点的扩散通量 J 以不同的速率、但在相同的时间都将会到达 0；
- (2) 空间中所有点的扩散通量随时间的变化率都是先快后慢的，也就是 $\frac{\partial J}{\partial t}$ 的绝对值由大变小；

(3) 空间中任意一点，当扩散通量 J 为 0 时，扩散通量随时间的变化率 $\frac{\partial J}{\partial t}$ 也为 0，反之亦然。

根据上述扩散通量 J 的这种特点，由于 $\frac{\partial J}{\partial t}$ 在空间和时间上均无相关公式定律支撑，本文再次采用 3.3.1 建立模型所述的拟合函数法。

空间上采用的是较为通用的 n 次多项式拟合公式：

$$P_n(x) = a_1x^n + a_2x^{n-1} + \dots + a_nx + a_{n+1} \quad (39)$$

时间上可采用的函数曲线有：负指数幂函数 $y = \frac{1}{x^n}$ 、自然对数的指数函数 $y = e^{-x}$ ，如图 8 所示，以及一类特殊的函数——最速降线。

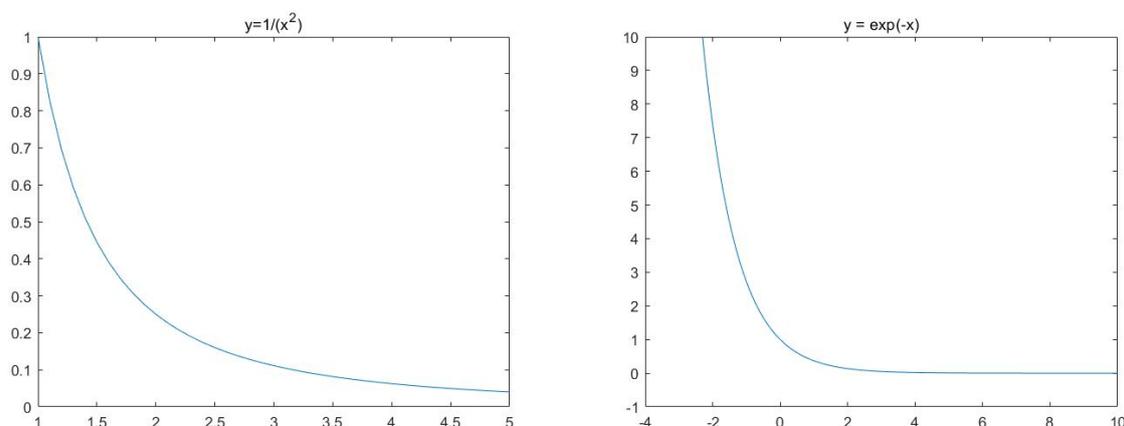


图8 随自变量增大因变量趋近于 0 拟合函数

其中，最速降线是经典的变分问题之一，该曲线有两个特点：一是该曲线是从某一空间上的较高点到较低点用时最短的曲线，二是该曲线上所有的点到达最低点的时间相同，不随空间分布而变化。

最速降线的表达式为：

$$\begin{cases} y = a(\theta - \sin \theta) \\ x = a(1 - \cos \theta) \end{cases} \quad (40)$$

由于这是非线性方程，求取最速降线首先通过对式 (40) 消元 a 得到关于形参 θ 的关系式，然后采用了牛顿迭代法（详见 3.2.3 节）求取 θ 的数值解，进而代入式 (40) 获得 a ，至此最速降线的表达式得以确定。

为了将扩散通量随时间的变化率 $\frac{\partial J}{\partial t}$ 与位置 x 的空间特性曲线随时间变化起来，本文通过将 $\frac{\partial J}{\partial t}$ 的空间拟合函数的系数与时间拟合函数相关联——因为所有位置的 $\frac{\partial J}{\partial t}$ 趋近于 0 也意味着各个多项式系数也全部趋近于 0，进而得到如下拟合函数：

$$\frac{\partial J}{\partial t} = P_n(x, t) = a_1(t)x^n + a_2(t)x^{n-1} + \dots + a_n(t)x + a_{n+1}(t) \quad (41)$$

通过上述对 $\frac{\partial J}{\partial t}$ 建立拟合函数的数学模型实现合理地预测扩散通量 J 的变化规律，进而推演出后续不同 T 时刻的 $\frac{\partial J}{\partial x}\Big|_{t=t_i}$ ，利用 Fick 第二定律将 $\frac{\partial C}{\partial t}\Big|_{x=x_i}$ 相应可以获得，进而通过时间积分获得组元浓度 C 在不同 T 时刻的变化量，当时间步数 N 取值足够大时，每一个位置的组元浓度 C 的变化量将会分别达到一个稳定的状态，即：

$$\lim_{t \rightarrow \infty} \int_0^t \left(\frac{\partial C}{\partial t} \right) dt = \lim_{t \rightarrow \infty} \int_0^t \left(-\frac{\partial J}{\partial x} \right) dt \approx \sum_1^N \left(-\frac{\partial J}{\partial x} \right) \Delta t \quad (42)$$

理论上这就意味着这是每个位置的组元浓度 C 初值到达稳定水平的变化量，因此当确定一个合适的稳定水平的大小 C_{end} 后，组元浓度 C 的初值 $C|_{t=0}$ 就能通过上述方式反算得到：

$$C|_{t=0} = C_{end} - \lim_{t \rightarrow \infty} \int_0^t \left(\frac{\partial C}{\partial t} \right) dt \approx C_{end} - \sum_1^N \left(-\frac{\partial J}{\partial x} \right) \Delta t \quad (43)$$

至此，将上述求得组元浓度 C 的理论值和扩散通量 J 的测量值作为数据基础，此时便可以使用上一节 3.3.1 所采用的扩散通量和组元浓度均已知的扩散系数识别模型进行扩散系数 D 的理论值计算，计算方法详见 3.3.1。

3.3.4 扩散系数识别精度分析模型

(1) 误差水平

建立扩散系数的识别模型后，需要对模型的识别精度进行分析。对组元浓度 C 以及扩散通量 J 的实测值人为地加入随机测量误差 ε 后，再使用本文 3.3.1 的扩散系数识别模型，得到新的扩散系数 D 的预估值。通过对比加入随机测量误差 ε 前后所得结果来分析模型的识别精度。

加入随机误差后的数据变为

$$\begin{aligned} \hat{J} &= J^* + \varepsilon \cdot k \cdot J^* \\ \hat{C} &= C^* + \varepsilon \cdot k \cdot C^* \end{aligned} \quad (44)$$

其中 J^* 与 C^* 为实测值（即附件 1、2 中的数据值），随机测量误差 ε 服从正态分布 $\varepsilon \sim N(0,1)$ ，且 k 分别取值为 0.1%、0.2%、0.3%、0.4%、0.5%。

(2) 数据量

数据的质量和数量通常是决定一个反问题求解模型性能较为关键的参数。在应用曲线拟合方法处理相同的问题时，数据量太少会导致模型无法完成充分的拟合，导致模型对新数据的预测结果差；当数据量过多会导致模型出现过度拟合的情况，使得模型准确性差。

本文采用均匀分布采样减少数据量，即经过固定的 k 个数据点后采一个数据点，根据附件数据特点 k 选取 100 的因数，分别就 $k=2、4、5、10$ 进行采样，这样的样本分布均匀，以保证位置 x 的步长 Δx 值是固定地变化为原来的 2、4、5、10 倍，因此得到数据量分别为 51、26、21、11 的数据集。

增加数据量则采用三次样条插值算法（详见 3.2.4），在原始数据中相邻两个点插入 k 个点以得到更多数据，分别就 $k=1、2、3、4、5、6、7、8、9$ 进行三次样条插值，使得位置 x 的步长 Δx 值是固定地变化为原来的 1/2、1/3、1/4、1/5、1/6、1/7、1/8、1/9、1/10，因此得到数据量分别为 201、301、401、501、601、701、801、901、101 的

数据集。

将减少或扩充后的数据集作为输入量，应用已有的 3.3.1 方法进行扩散系数 D 的计算，并分析计算精度。

3.3.5 多组元高维扩散过程扩散系数识别模型

对于多组分物系，由于组分之间存在交互作用，扩散系数的计算相当复杂。由于实际发生的传质过程基本都在多组分体系中发生，因此研究多组元高维扩散过程扩散系数具有重要的实际意义。为了建立扩散系数识别模型，首先要将适用于单组分与二组分物系的 Fick 扩散定律推广到多组分体系。则 Fick 第一定律扩展为

$$J = -\hat{D}\nabla C$$

其中 \hat{D} 为适用于多组分物系的 Fick 扩散系数矩阵

$$\hat{D} = \begin{bmatrix} D_{11} & D_{12} & \dots & D_{1n} \\ D_{21} & D_{22} & \dots & D_{2n} \\ \vdots & & & \vdots \\ D_{n1} & D_{n2} & \dots & D_{nn} \end{bmatrix}$$

C 为多组元体积浓度，表示为

$$C = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & & & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix}$$

$$\nabla C = \frac{\partial C}{\partial x} + \frac{\partial C}{\partial y} + \frac{\partial C}{\partial z}$$

最终多组分体系 Fick 第一定律可以表示为

$$J = - \begin{bmatrix} D_{11} & D_{12} & \dots & D_{1n} \\ D_{21} & D_{22} & \dots & D_{2n} \\ \vdots & & & \vdots \\ D_{n1} & D_{n2} & \dots & D_{nn} \end{bmatrix} \cdot \left(\frac{\partial C}{\partial x} + \frac{\partial C}{\partial y} + \frac{\partial C}{\partial z} \right)$$

组元高维扩散过程扩散系数可表示为

$$\begin{bmatrix} D_{11} & D_{12} & \dots & D_{1n} \\ D_{21} & D_{22} & \dots & D_{2n} \\ \vdots & & & \vdots \\ D_{n1} & D_{n2} & \dots & D_{nn} \end{bmatrix} = - \frac{J}{\left(\frac{\partial C}{\partial x} + \frac{\partial C}{\partial y} + \frac{\partial C}{\partial z} \right)}$$

相应地将 3.3.1 节所使用的扩散通量和组元浓度均已知的扩散系数识别模型方法进行矩阵延拓，理论上就能使用数值求解得到组元高维扩散过程扩散系数。

3.4 模型效果指标定义

本文中模型效果指标包含精度评价指标、稳定性指标和计算效率指标。

3.4.1 精度评价指标

本文的核心是通过数据得到反演扩散系数的回归模型，扩散系数的计算精度表现为该模型的精度。常用的精度评价指标包括平均绝对误差、均方误差、均方根误差，

可以准确地计算出计算结果与真实结果之间的差值。而本文中并未给出扩散系数的真值，只能以测量数据间接所得扩散系数为真值。故文中的精度评价指标可定义为。

$$MSE_{D_{err}} = \frac{1}{m} \sum_{i=1}^m (D_i - D_i^*)^2 \quad (45)$$

其中 $MSE_{D_{err}}$ 表示扩散系数计算精度， m 表示数据量的大小，下标“ i ”表示第 i 个数据点，“*”号表示根据原始数据所得扩散系数，不带“*”号表示根据文中所建立的模型计算所得扩散系数。该指标越小意味着计算精度越高。

3.4.2 稳定性指标

算法的稳定性体现在计算结果的波动程度上，波动程度越大则算法越不稳定，本文通过方差对算法的稳定性进行评估，则稳定性指标定义为

$$S_D = \frac{1}{m} \sum_{i=1}^m (D_i - \bar{D})^2 \quad (46)$$

其中， S_D 表示文中算法的稳定性指标， \bar{D} 表示扩散系数计算结果的平均值。该指标越小意味着算法越稳定，反之则越不稳定。

3.4.3 计算效率指标

考虑到由于计算机硬件和编程语言都有可能会对计算效率产生影响，本文在第二章模型假设下只考虑算法本身带来的效率问题。目前基本上所有的扩散系数反演计算问题包含有差分格式计算、梯度计算等需要计算偏导数的过程。可以认为，算法的计算效率体现在算法对数值梯度的依赖程度，求解过程中需要用到的数值梯度越多，则计算效率越低，反之则反映了一个算法较高的计算效率。故计算效率指标（Calculation Efficiency Index, CEI）可定义为一个与数值梯度计算数量成正比的数。实际衡量过程中，可将数值梯度计算量直接作为衡量计算效率的指标。

4 模型应用及分析

4.1 已知扩散通量和组元浓度的扩散系数计算

应用 3.3.1 节中的扩散系数测量值计算，首先通过前向差分或中心差分计算所得扩散系数与浓度之间的关系如图 9 和图 10 所示。

图 9 给出了采用前向差分公式计算所得扩散系数测量值 D 在体积浓度 C 上的分布特征。扩散系数 D 在 $C = 0.01 \sim 0.06$ 之间的分布近似呈现一条平行于横轴的直线，幅值震荡很微弱，不同 T 时刻的曲线普遍在 $C < 0.01$ 和 $C > 0.06$ 时，扩散系数 D 的曲线出现大幅震荡。这是因为在数值计算中接近边界的位置分数项的分子和分母均接近于 0，二者相除所得的结果容易因为数据的误差发生骤增或骤减，因此越接近边界其震荡幅度就越大。

图 10 给出了采用中心差分公式计算所得扩散系数测量值 D 在体积浓度 C 上的分布特征。因为与前向差分使用的是相同的数据集，因此结果与前向差分所得分布趋势基本相同，但振荡程度更低一些。为了保证后续计算的稳定性，本文后续将采用中心差分方式对数值梯度进行计算。

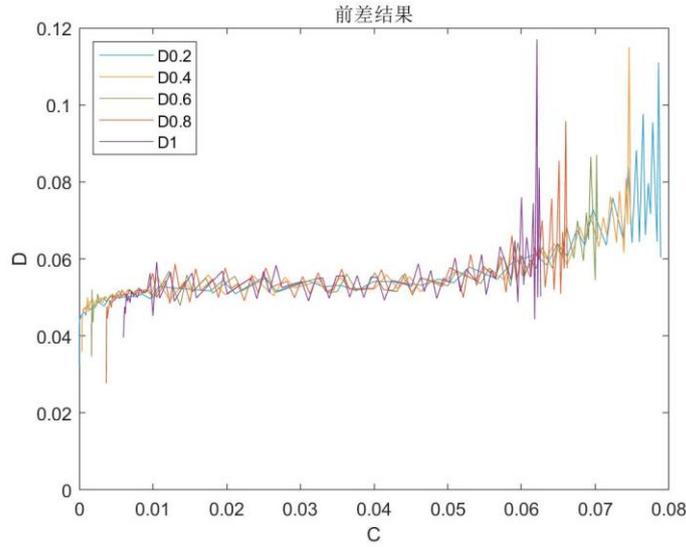


图9 前向差分公式计算所得扩散系数测量值 D

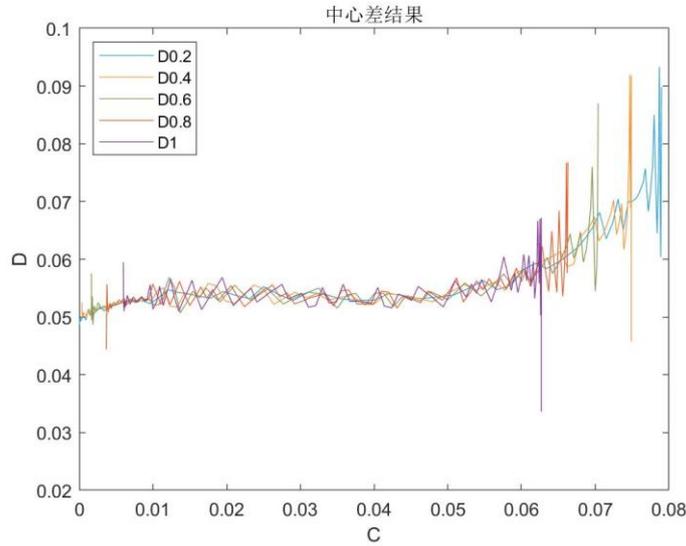


图10 中心差分公式计算所得扩散系数测量值 D

为了计算扩散系数 D ，需要得到浓度 C 关于空间的解析式，解析式可由拟合得到。接下来确定拟合函数的选取，因为任意时刻浓度 C 的空间分布特征基本类似，所以选取其中 $T=0.2$ 时刻的数据点，三种函数的拟合效果如图 11 所示。

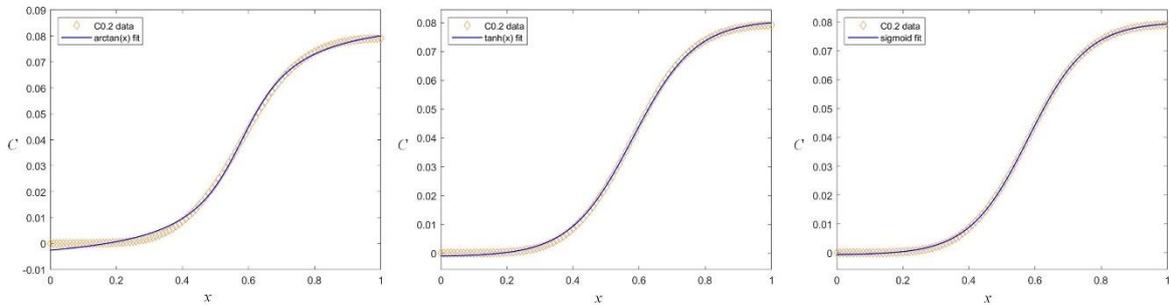


图11 反正切函数、双曲正切函数和 sigmoid 函数在 $T=0.2$ 时刻的拟合效果示意图

根据上图可以看出三种函数均能够大致拟合出浓度关于空间的关系式，但是不同函数仍存在一些精度上的差异，三者比较异同如图 12 所示。

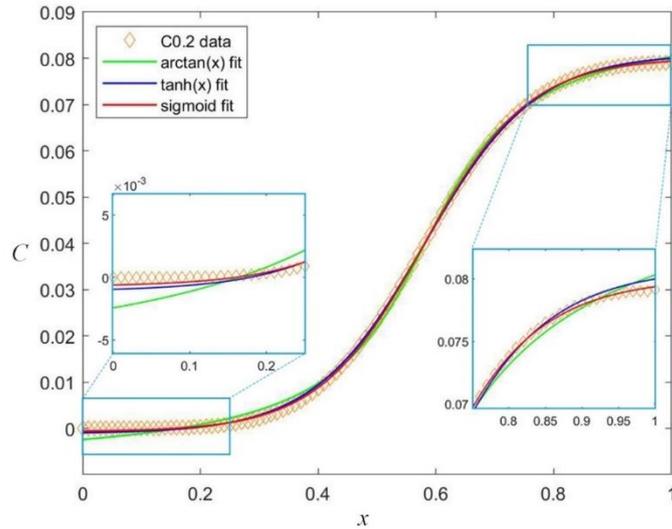


图12 三种拟合函数在拟合效果对比

为进一步定量分析三种拟合函数的拟合效果，三种拟合函数在 $T=0.2$ 时刻拟合结果的拟合误差计算如表 2 所示，同时表中给出了确定拟合函数（详见式 (26)）具体表达的 4 个参数。

表2 $T=0.2$ 时刻拟合函数参数与误差表

C0.2	arctan(x)	tanh(x)	sigmoid
a₁	0.03131	0.04102	0.0808
a_{2,1}	8.0495	5.3099	11.199
a_{2,2}	0.5824	0.5806	0.5794
a₃	0.04017	0.0399	-0.00072
Error	1.22×10^{-4}	2.96×10^{-5}	8.36×10^{-6}

其中，拟合误差 Error 通过均方误差计算，方法与 3.4.1 节中精度评价指标类似。由表 2 可见，反正切函数的拟合效果较差，而 sigmoid 函数拟合精度比双曲正切函数高一个数量级。故本文选择 sigmoid 函数作为浓度 C 在空间分布上的拟合函数。

在不同时刻确定拟合函数具体形式的 4 个参数如表 3 所示。

表3 不同时刻确定 sigmoid 拟合函数的四个参数值及对应拟合误差

sigmoid	a₁	a_{2,1}	a_{2,2}	a₃	Error
C0.2	0.0808	11.199	0.5794	-7.23×10^{-4}	8.36×10^{-6}
C0.4	0.079	8.1198	0.5695	-9.00×10^{-4}	6.04×10^{-6}
C0.6	0.07623	6.7198	0.5564	-7.90×10^{-4}	1.07×10^{-5}
C0.8	0.0714	6.1099	0.543	4.03×10^{-4}	1.08×10^{-5}
C1	0.06565	5.8198	0.5314	0.00234	9.30×10^{-6}

从表中可以看出，不同时刻 sigmoid 函数的拟合误差均达到了 10^{-5} 量级，验证了 sigmoid 函数具有良好的拟合效果。

对拟合函数求导，结合扩散通量的测量值，即可得到扩散系数的理论计算值，如图 13 所示。

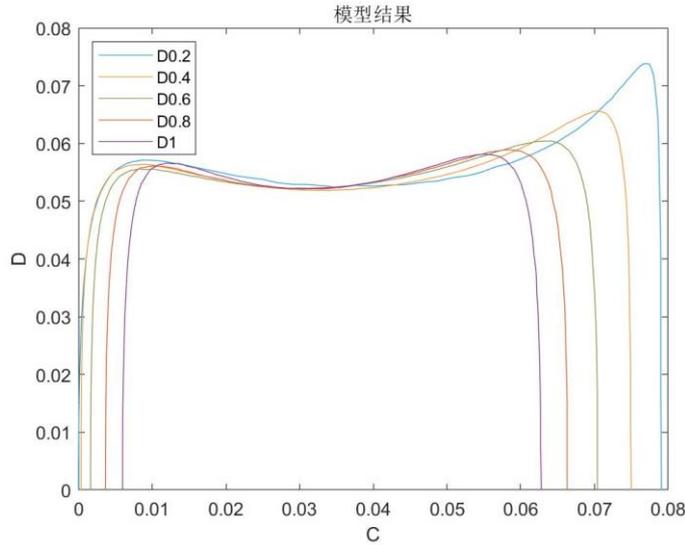


图13 模型计算所得扩散系数 D 与浓度 C 之间的关系

图 13 所示的扩散系数 D 和组元浓度 C 之间的关系与图 10 中经过中心差分计算所得结果在趋势上基本吻合：当浓度 $C = 0.01 \sim 0.06$ 时，扩散系数 D 接近于一个常值，当 $C > 0.06$ 之后 D 呈现上升趋势。但是两端的计算结果具有显著差异，此现象的原因在 3.3.1 中的扩散系数-组元浓度曲线中已有分析：在测量条件下，边界处的扩散通量 J 和浓度梯度 $\frac{\partial C}{\partial x}$ 均无限趋近 0，两者的比值为 0 与 0 的比值，该比值会出现振荡现象，而根据本文的模型计算出的浓度梯度 $\frac{\partial C}{\partial x}$ 是一个非 0 的值，则计算出来的扩散系数 D 会在边界处骤减为 0。

进一步分析计算所得经过组元浓度 C 数据归一化之后的扩散系数 D 与测量值之间的差异如图 14 所示。图中“model”表示本文模型计算所得扩散系数，“data”表示根据测量数据所得扩散系数。从图 14 中可见，根据本文模型计算所得扩散系数 D 与测量值随浓度 C 的变化趋势基本吻合，但是在浓度的两端出现了较大的差异，具体误差如图 15 所示。

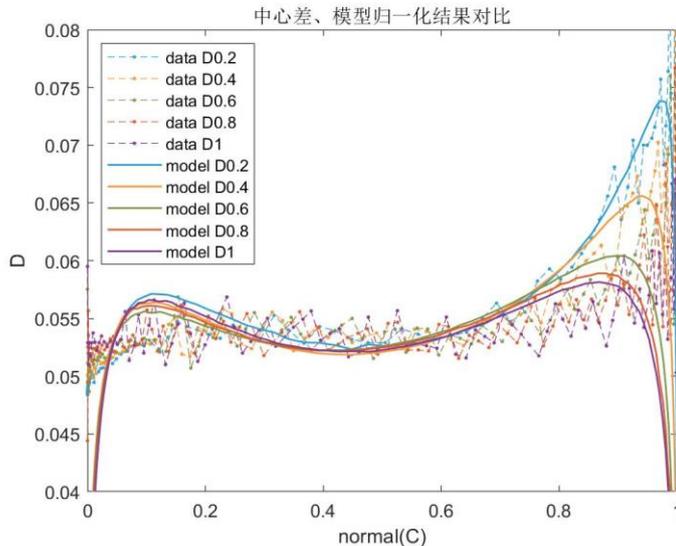


图14 浓度归一化后扩散系数 D 测量值与计算值的结果对比

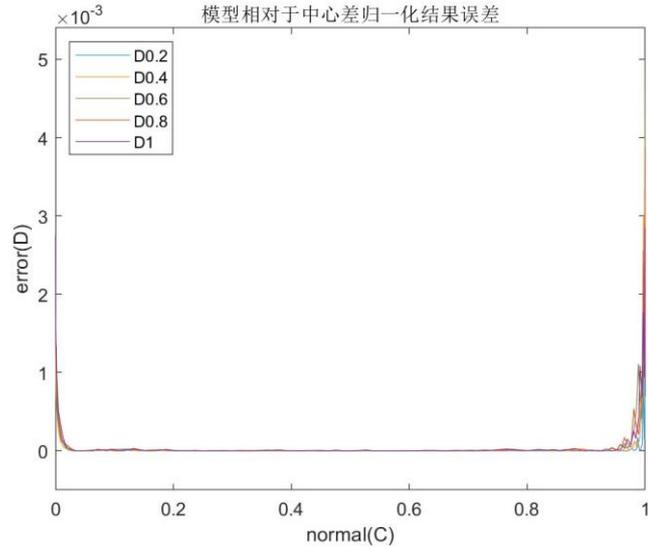


图15 模型修正前扩散系数 D 计算结果误差

为了使模型预测精度更高，使用式 (32) 的方法对模型进行修正，主要提高了边界处的预测精度。计算所得扩散系数与经过中心差分计算所得扩散系数测量值对比如图 16 所示。可以发现，修正后的模型预测值在两端与原始数据趋势更加符合，肉眼可见的提高了模型的准确度。

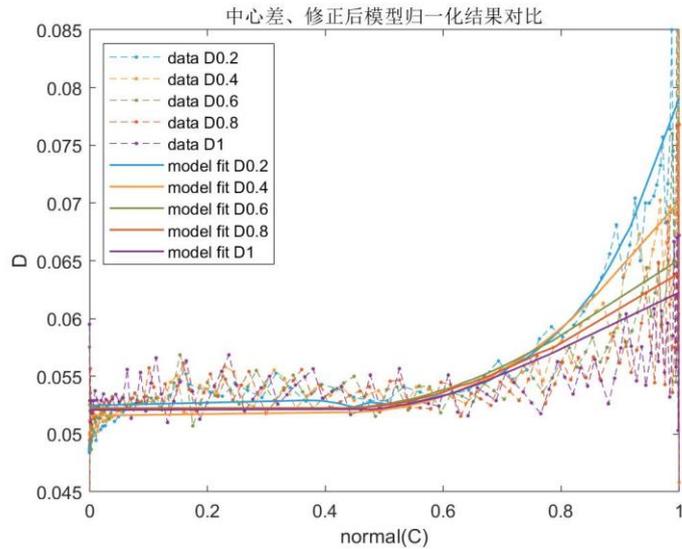


图16 模型修正后扩散系数 D 与测量值的对比

图 17 对比了模型修正前后，随浓度 C 变化的扩散系数 D 的曲线。其中，“model”表示修正前根据模型计算所得扩散 D 系数；“model fit”表示模型修正后扩散系数 D 的计算值。

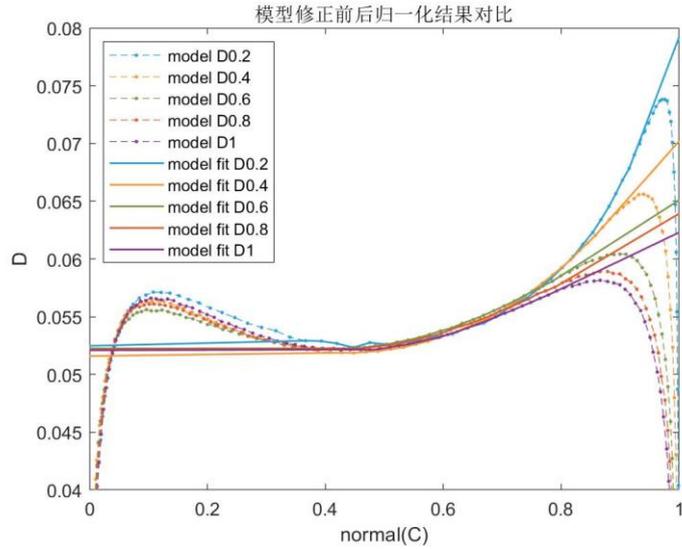


图17 组元浓度 C 归一化处理修正前后扩散系数结果对比

修正后扩散系数 D 计算值与测量值之间的计算误差如图 18 所示。可以看出，修正后的模型对扩散系数进行计算时能保持中间部分误差极小，并能将靠近两端的计算误差减小一个数量级。验证了引入修正项的正确性。修正后扩散系数与浓度的关系如图 19 所示。

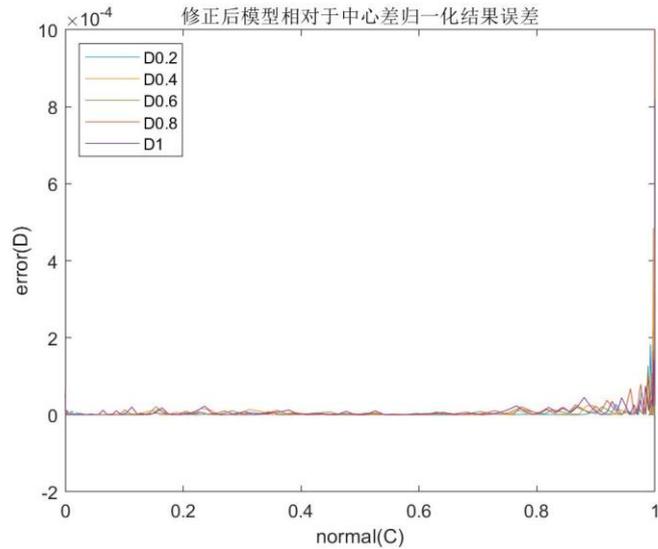


图18 模型修正后计算所得扩散系数 D 与测量值的误差

图 19 给出了模型修正后，不同时刻随浓度 C 变化的扩散系数 D 的曲线，即为本文给出的问题一的结果。

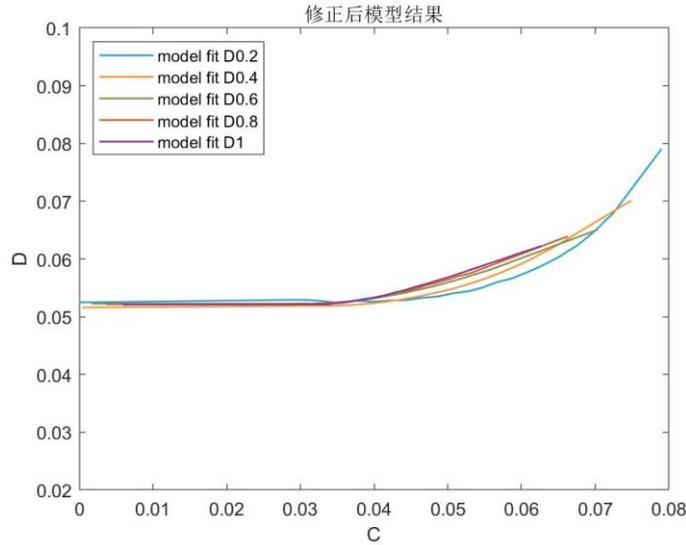


图19 模型修正后计算所得扩散系数 D

4.2 已知组元浓度的扩散系数计算

组元浓度 C 已知，扩散通量 J 未知的情况下，首先需要充分利用已知条件，对组元浓度 C 开展研究。组元浓度 C 在时间 t 与空间 x 上的分布如图 20 所示。 $T=0$ 时刻，浓度 C 在空间中变化不连续，在其余 5 个时刻中，浓度 C 的空间分布趋势相近，均呈“S”型分布。

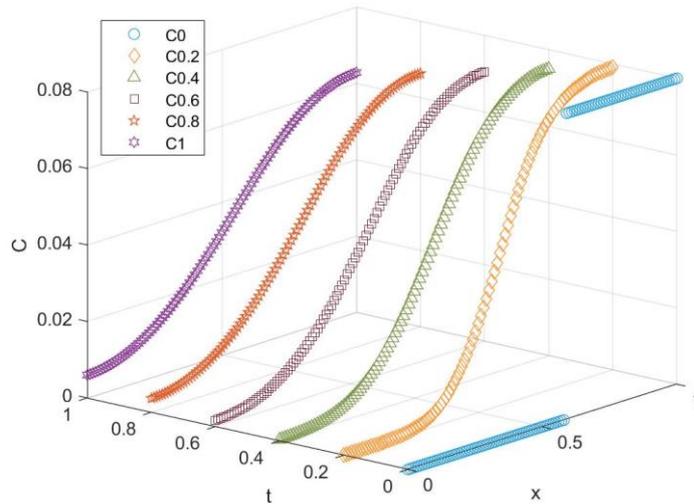


图20 组元浓度 C 在时空中的分布图

根据实验数据得到浓度 C 随时间的变化率 $\frac{\partial C}{\partial t}$ 的相反数在空间上的变化趋势如图 21 所示。可以发现，随着时间的增加，变化率的分布振幅逐渐减小，最终趋于一条直线。

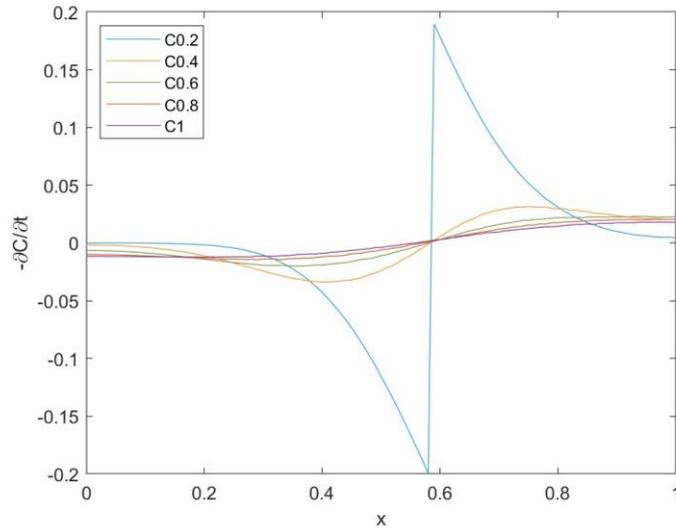


图21 浓度随时间变化率的相反数与空间位置的关系

根据 3.3.2 所建立的模型计算所得扩散通量与问题一中提供的扩散通量 J 对比如图 22 所示，其中曲线代表模型计算所得曲线，不同形状的点代表问题一中提供的原始数据。根据分布图像可以发现，本题模型计算所得 J 与原始值存在一定误差。在 $t=0.2$ 时刻误差最大，这是由于 $T=0$ 时刻浓度 C 的分布存在间断，导致浓度 C 随时间的变化率 $\frac{\partial C}{\partial t}$ 不连续。随着时间的增大，使用模型计算所得 J 与原始数据的分布逐渐吻合。

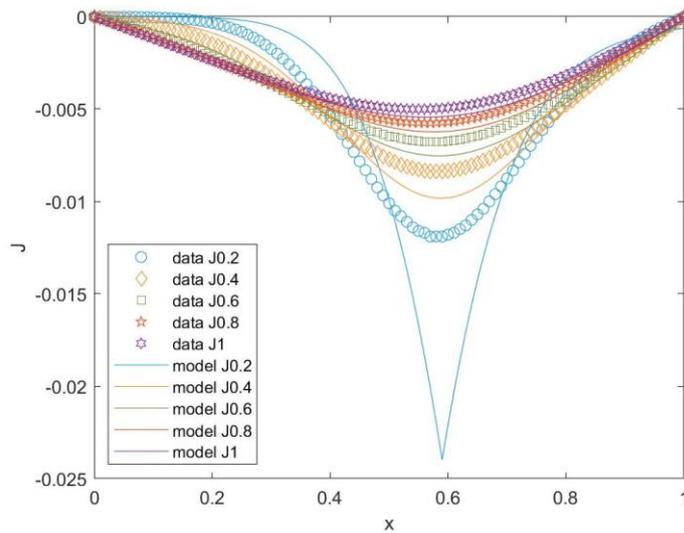


图22 使用模型所得扩散通量 J 在空间位置上的分布与问题一中测量数据的对比

基于以上数据，使用本题模型可以求出扩散系数 D 随浓度 C 变化的曲线。图 23 将问题一和问题二中不同模型（未修正）计算所得 D 随 C 变化的曲线进行对比，其中 Q1 model fit 表示使用问题一中模型所得曲线，Q2 model fit 表示使用本题建立模型所得曲线。可以发现，除了 $t=0.2$ 时刻的变化曲线以外，其余相同时刻的曲线变化趋势相近，但仍存在着不小的误差。

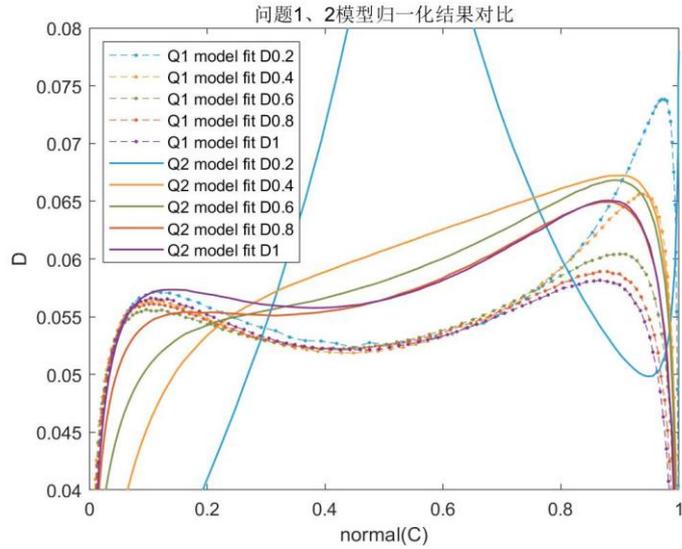


图23 问题一和 3.3.2 节计算所得扩散系数 D 随浓度 C 变化曲线的对比

将本题模型进行修正得到图 24 与图 25 中扩散系数 D 随浓度 C 变化的曲线，并将其与问题一中模型修正后所得数据进行对比，图 24 中的曲线做了归一化处理，其中 Q1 model fit 表示使用问题一中模型所得曲线，Q2 model fit 表示使用本题建立模型所得曲线。两种模型所得曲线变化趋势相近，幅值差距在可接受的范围内，说明本题模型是可靠的，可以得到较为准确的数据。

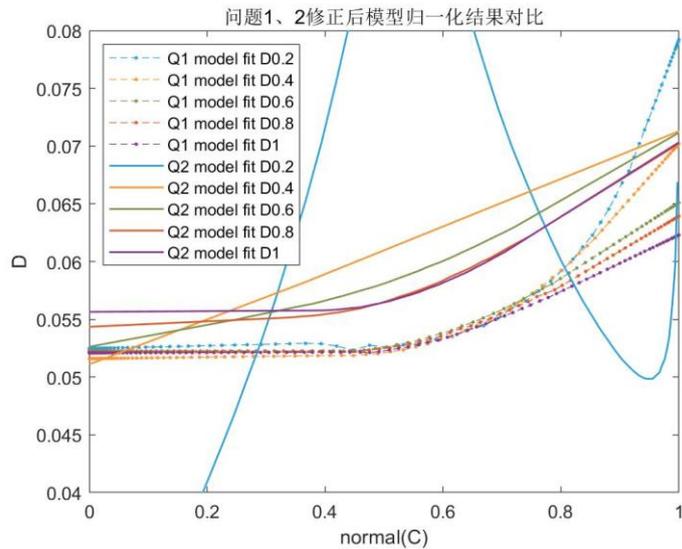


图24 3.3.2 节模型修正后的归一化扩散系数 D 计算结果与问题一结果的对比

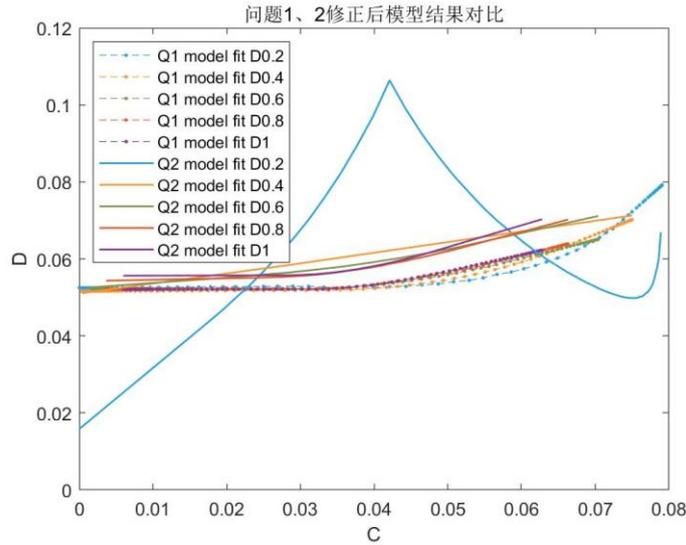


图25 3.3.2 节模型修正后扩散系数 D 计算值与问题一结果的对比

4.3 已知扩散通量的扩散系数计算

本题中扩散通量 J 已知，组元浓度 C 未知。扩散通量 J 在时间 t 与空间 x 上的分布如图 20 所示。 $t=0$ 时刻，扩散通量 J 在空间中恒为 0，在其余 5 个时刻中，浓度 C 的空间分布趋势相近，均呈现凹函数分布。

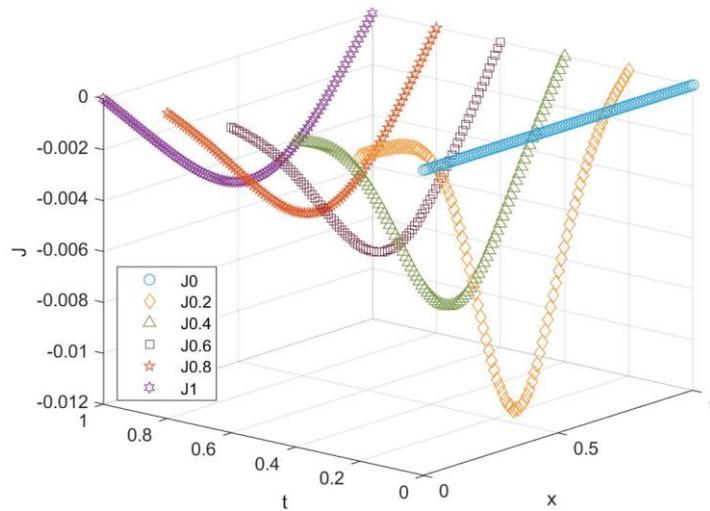


图26 扩散通量 J 在时空上的分布

通过不同时刻 T 的扩散通量的空间分布进一步可以得到后续不同 T 时刻的 $\frac{\partial J}{\partial x} \Big|_{t=t_i}$ ，如图 27 所示。可以发现，随着时间的推移，每一个位置的组元浓度 C 的变化量将会分别达到一个稳定的状态。

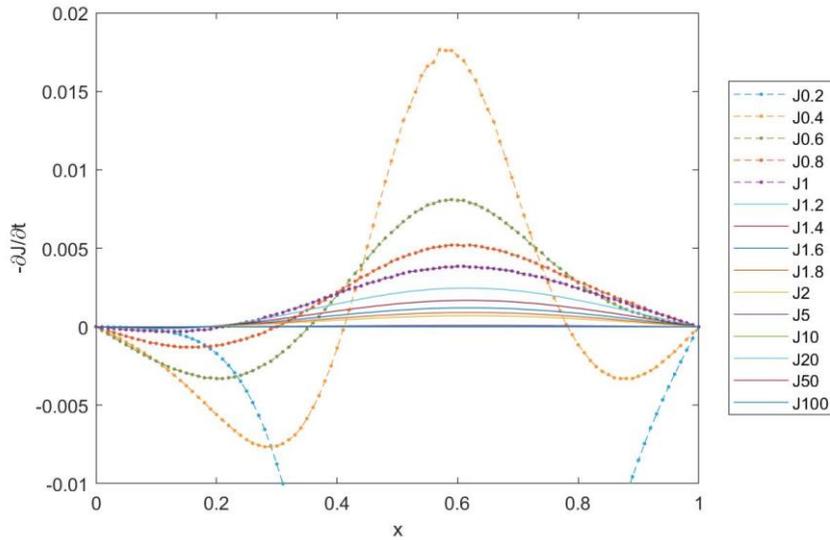


图27 不同 T 时刻的 $-\frac{\partial J}{\partial t}$ 在空间上的分布

根据 3.3.3 节的模型，对扩散通量时间变化率 $\frac{\partial J}{\partial t}$ 在空间上采用多项式拟合，根据拟合效果评估至少需要采用 4 阶及以上的多项式拟合效果会达到标准，因此本文结合边界条件采用 (4+2) 阶的多项式拟合，将式 (39) 实际化简为：

$$P_6(x) = x(x-1)(a_1x^4 + a_2x^3 + a_3x^2 + a_4x + a_5) \quad (47)$$

通过预估系数对时间 t 拟合函数，经试验使用负指数幂的拟合兼具计算效率和计算精度的要求，上述式 (47) 的预估系数随时间变化如图 28 所示。

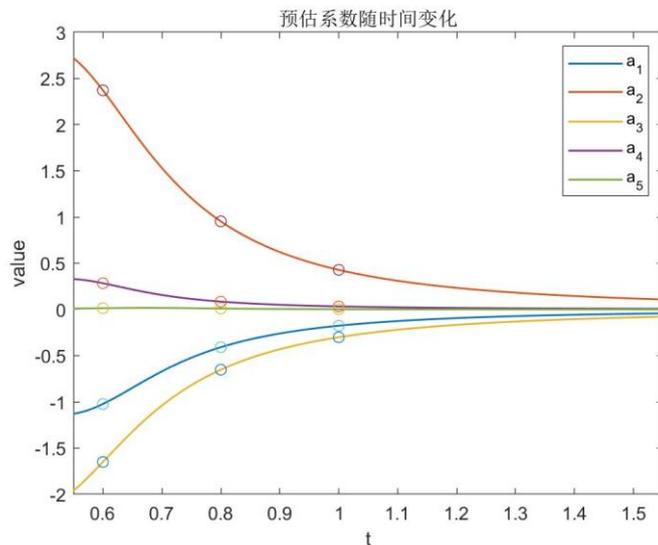


图28 预估系数随时间变化曲线

为了求解扩散系数 D ，还需要得到组元浓度 C 的理论计算值，但是根据 Fick 第二定律求解时，组元浓度 C 在时间上 $T=0$ 时刻的初值是未知的，因此无法通过积分得到组元浓度 C 的理论计算值。本题模型另辟蹊径，通过对 $\frac{\partial J}{\partial t}$ 建模预测，预测出后续不同 T 时刻的 J ，如图 29 所示。

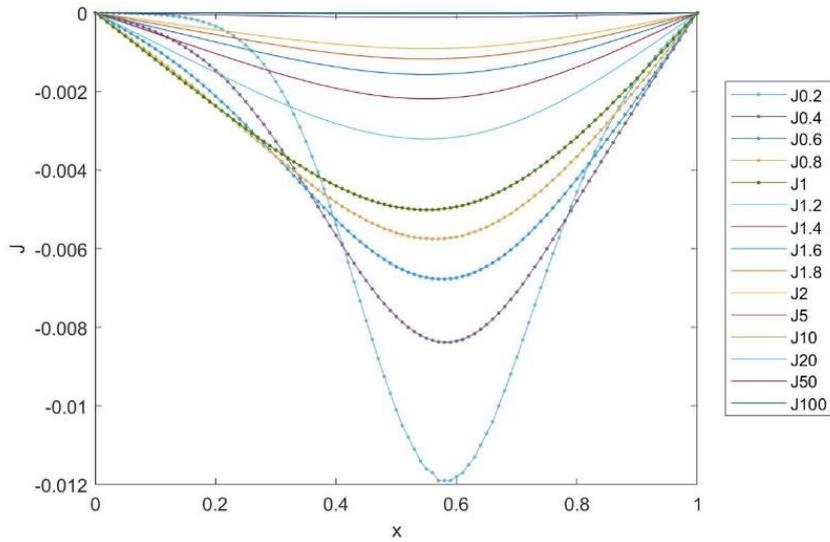


图29 不同时刻 T 的扩散通量的空间分布

已知扩散通量 J 的空间分布后，其空间梯度 $\frac{\partial J}{\partial x}$ 也可使用中心差分求出，如图 30 所示。进而根据 Fick 第二定律得出 $\frac{\partial C}{\partial t}$ ，通过时间积分可得组元浓度 C ，最后和问题一
中所得扩散系数 D 随浓度 C 变化曲线的对比如图 31 所示。

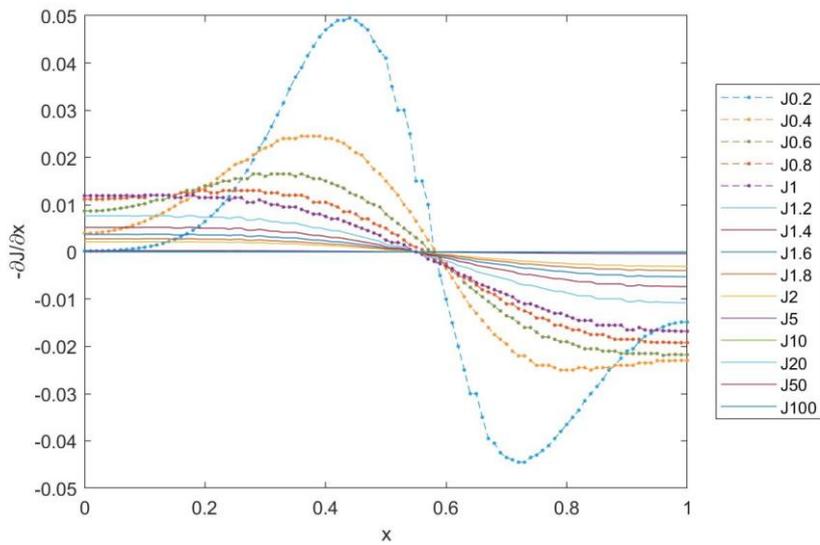


图30 不同时刻 T 的 $-\frac{\partial J}{\partial x}$ 在空间上的分布

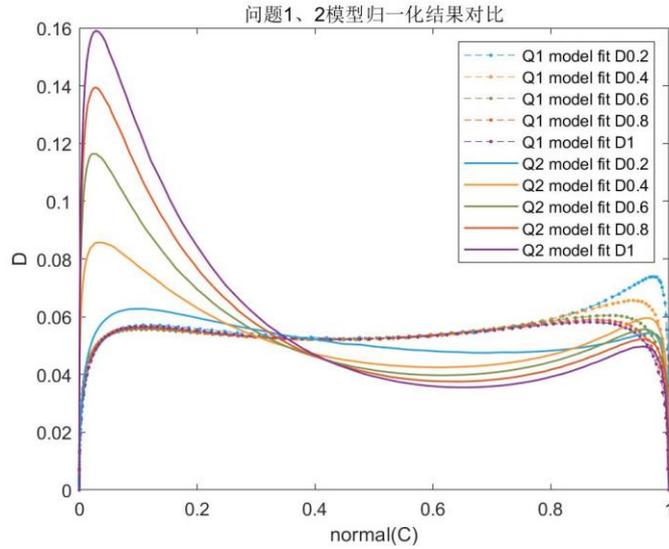


图31 3.3.3 节模型和问题一中所得扩散系数 D 随浓度 C 变化曲线的对比

图 32 和图 33 展示了使用经过修正后的 3.3.3 节模型所得结果，并将其与问题一中的曲线进行比较。Q1 model fit 表示使用问题一中模型所得曲线，Q2 model fit 表示使用 3.3.3 节模型所得的曲线。可以发现，本节计算所得扩散系数 D 的值相比问题一中所得 D 值整体偏小，且不同时刻 T 条件下的曲线分布较为分散，但随着时间的推移，曲线也表现出逐渐重叠的趋势。但是曲线整体的变化规律是相同的，说明本节模型能够得到有效可靠的数据。

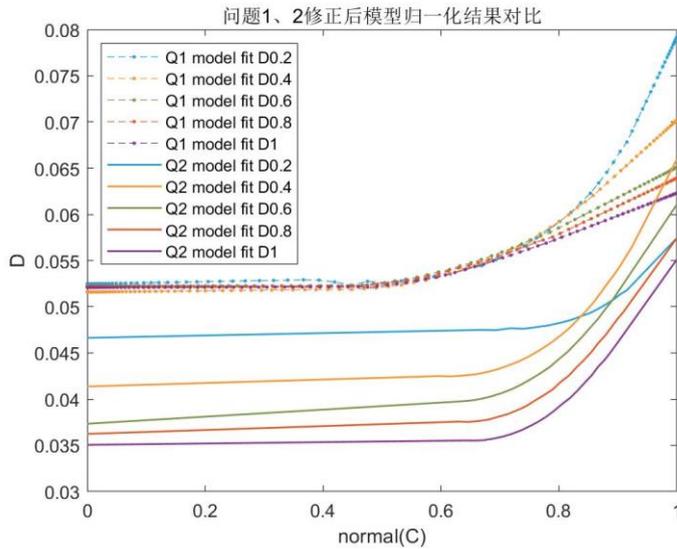


图32 3.3.3 节模型修正后的归一化扩散系数 D 计算结果与问题一结果的对比

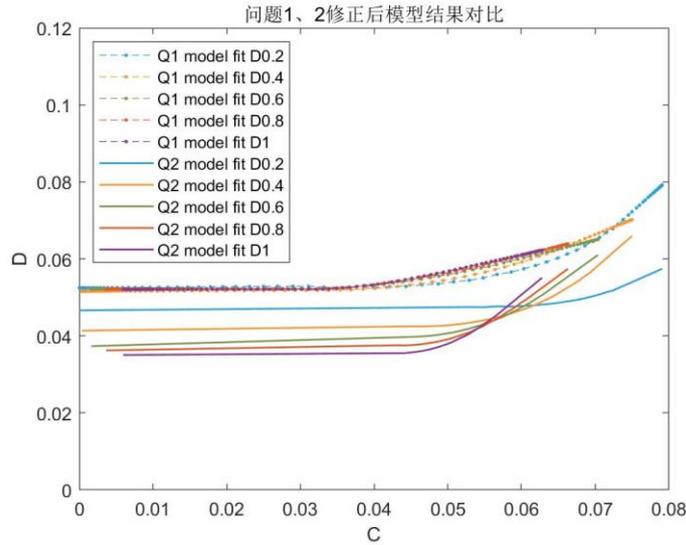


图33 3.3.3 节模型修正后扩散系数 D 计算值与问题一结果的对比

4.4 扩散系数识别精度分析

4.4.1 误差水平变化的影响

建立扩散系数的识别模型后，需要对模型的识别精度进行分析。对实测值加入随机误差后，再使用本文的扩散系数识别模型，得到扩散系数。本节根据 3.3.4 节的方法加入随机误差，实测值变化如图 28 所示，可以看到实测值的曲线发生了波动，不再光滑。图 34 图 35 图 36 图 37 图 38 图 39 分析了加入随机误差后，模型预测的精度以及得到的扩散系数 D 的分布。

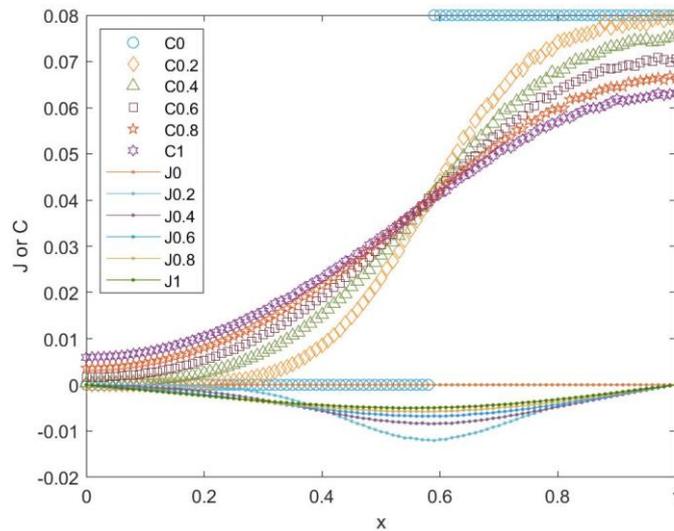


图34 增加随机误差的条件下，扩散通量 J 和浓度 C 的空间分布

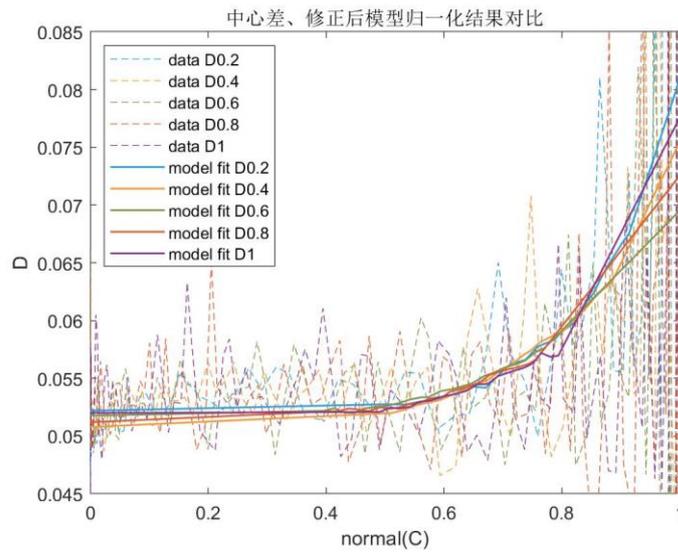


图35 增加随机误差的条件下，模型修正后与中心差分法的归一化扩散系数 D 分布对比

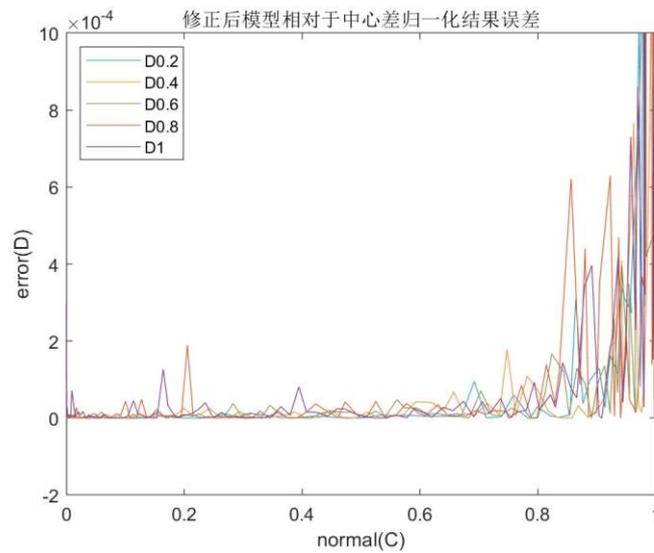


图36 增加随机误差的条件下，修正模型相对于中心差分法归一化结果的误差

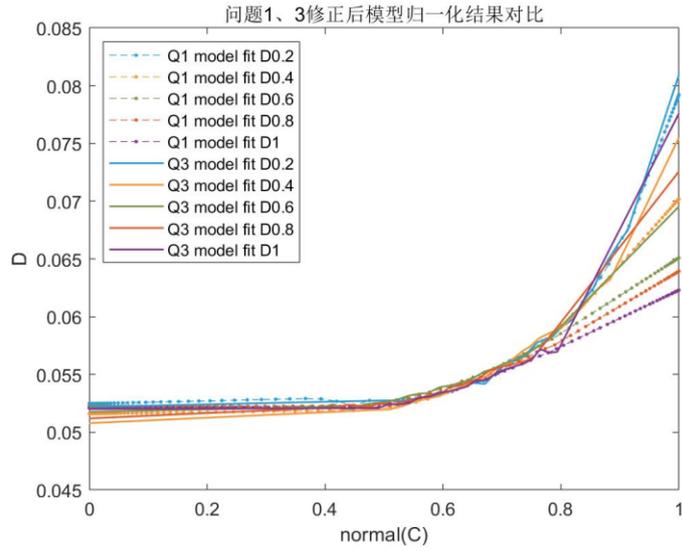


图37 增加随机误差的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

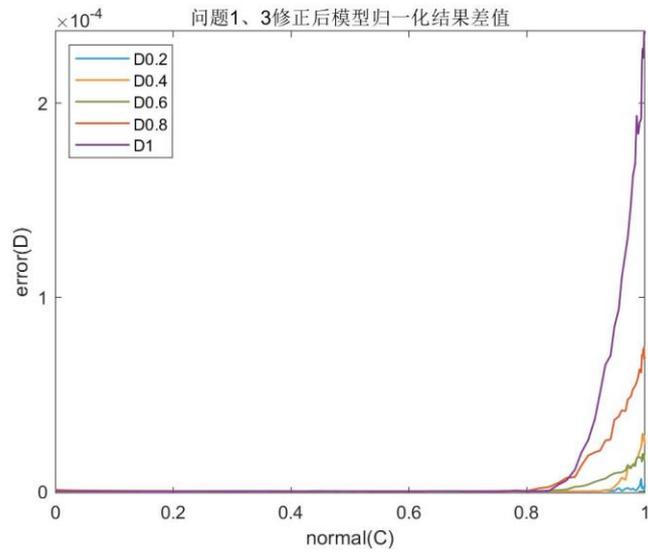


图38 增加随机误差的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的差值

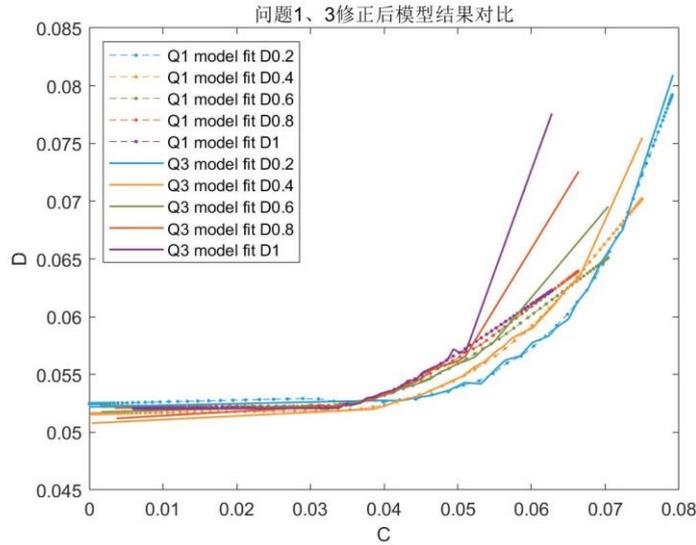


图39 增加随机误差的条件下，模型修正后的扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

下图 40 给出了加入随机误差后模型误差的变化情况，模型误差随着加入随机误差的增大而增大。

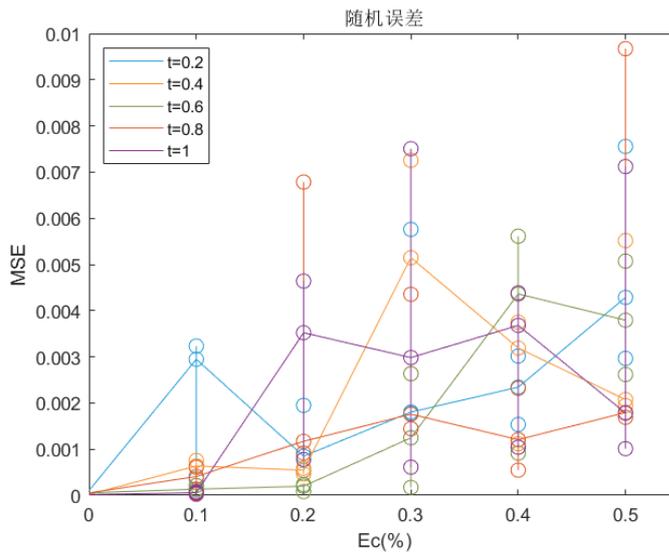


图40 随加入的随机误差变化的误差分布对比

4.4.2 数据量变化的影响

运用曲线拟合解决问题时，数据量的大小会对拟合效果产生重要影响。本节使用采用均匀分布采样减小数据量，以数据量大小为 11 举例，扩散通量 J 和浓度 C 的空间分布如图 41 所示。图 42 图 43 图 44 图 45 图 46 分析了减小数据量后，模型预测的精度以及得到的扩散系数 D 的分布。

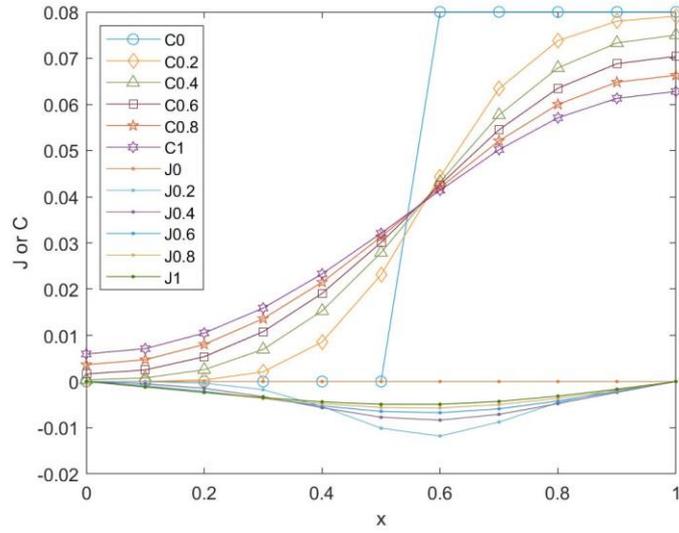


图41 数据量为 11 的条件下，扩散通量 J 和浓度 C 的空间分布

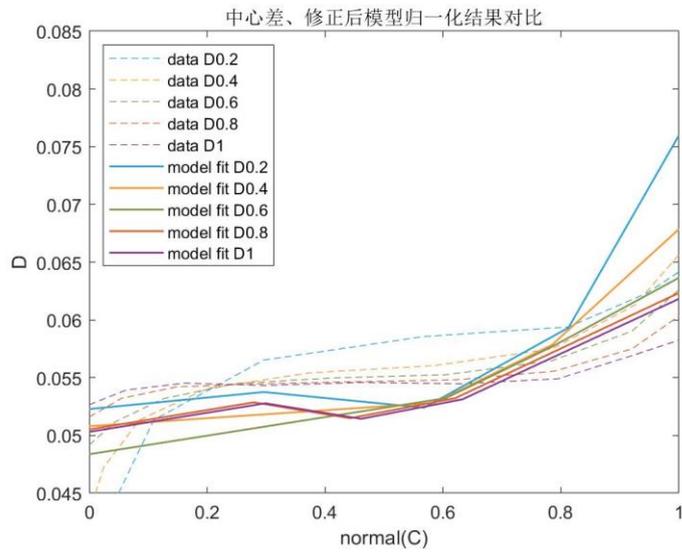


图42 数据量为 11 的条件下，模型修正后与中心差分法的归一化扩散系数 D 分布对比

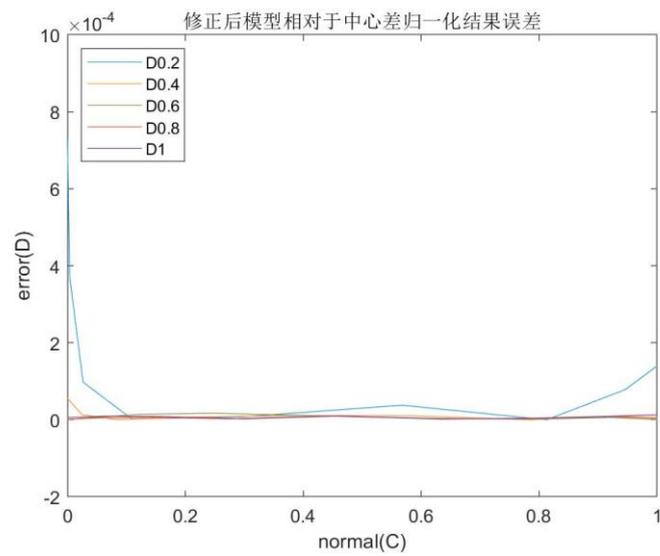


图43 数据量为 11 的条件下，修正模型相对于中心差分法归一化结果的误差

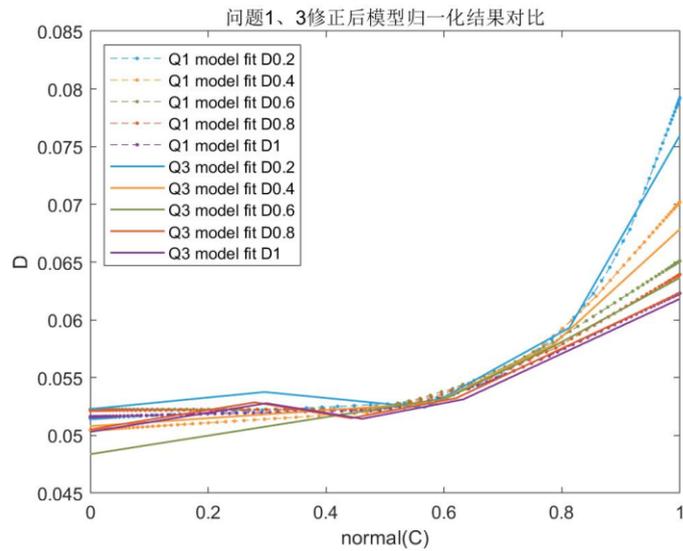


图44 数据量为 11 的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

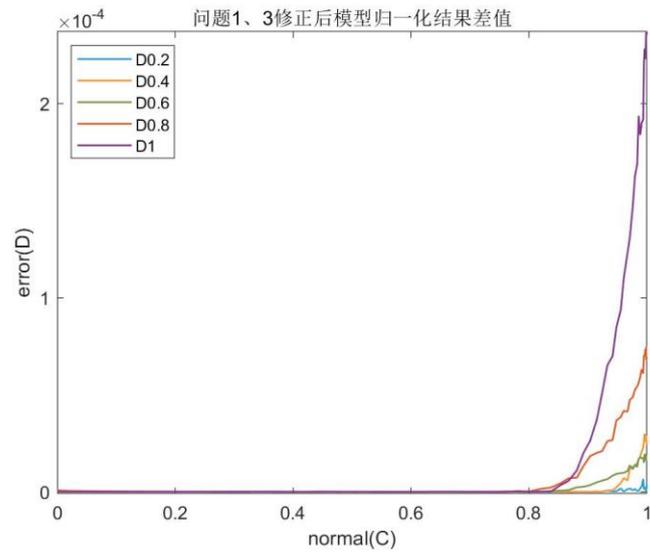


图45 数据量为 11 的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的差值

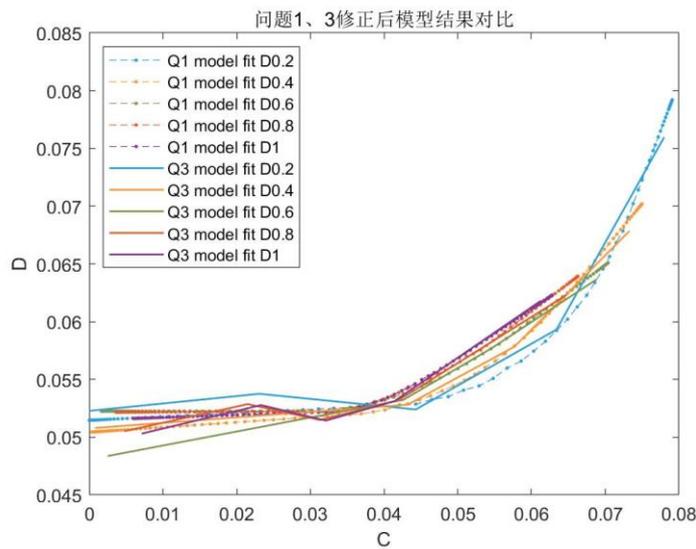


图46 数据量为 11 的条件下，模型修正后的扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

本节使用三次样条插值增大数据量，以数据量大小为 1001 举例，扩散通量 J 和浓度 C 的空间分布如图 47 所示。图 48 图 49 图 50 图 51 图 52 分析了增大数据量后，模型预测的精度以及得到的扩散系数 D 的分布。

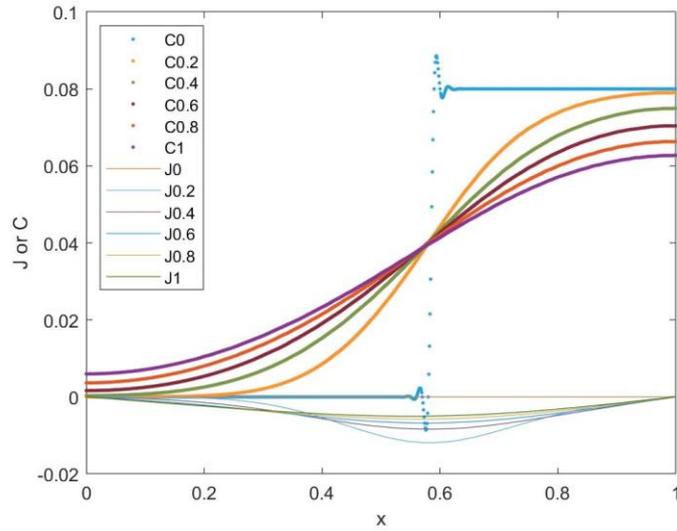


图47 数据量为 1001 的条件下，扩散通量 J 和浓度 C 的空间分布

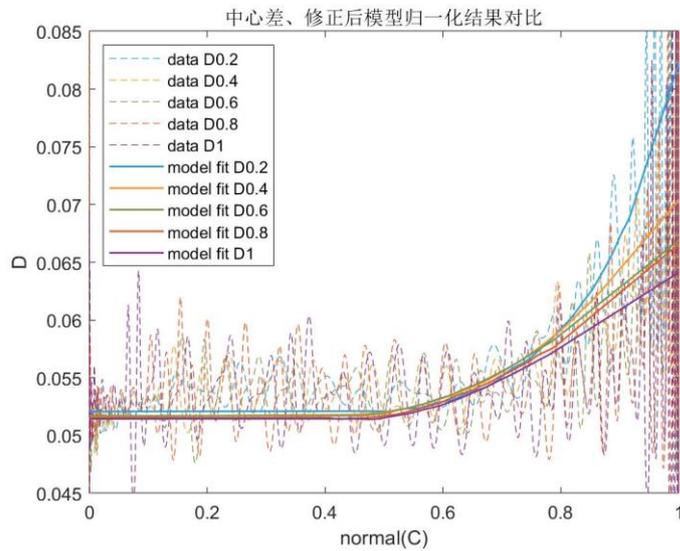


图48 数据量为 1001 的条件下，模型修正后与中心差分法的归一化扩散系数 D 分布对比

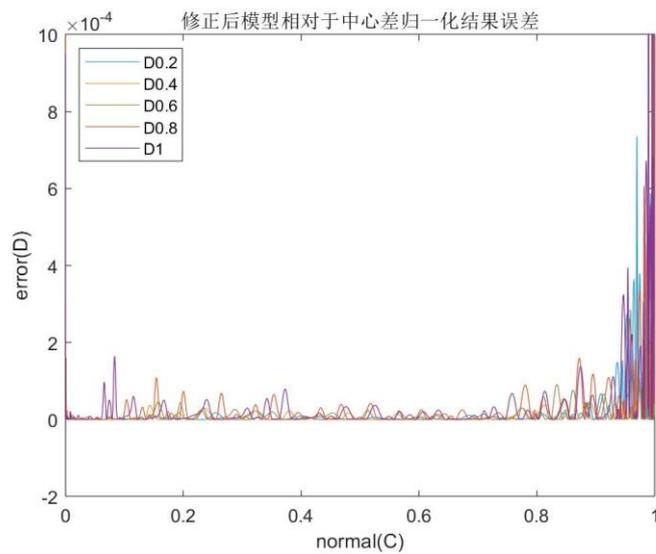


图49 数据量为 1001 的条件下，修正模型相对于中心差分法归一化结果的误差

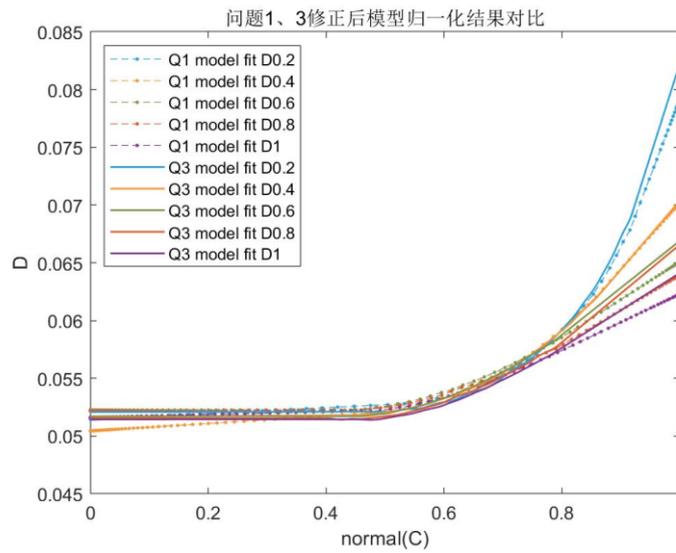


图50 数据量为 1001 的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

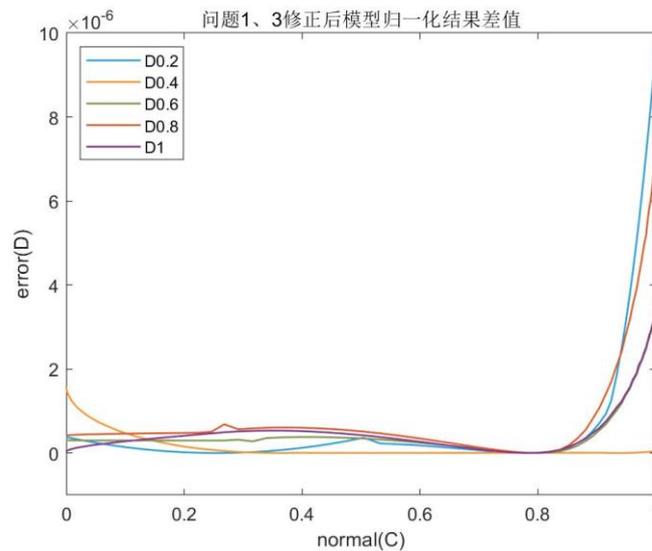


图51 数据量为 1001 的条件下，模型修正后的归一化扩散系数 D 计算结果与问题一得到的扩散系数 D 的差值

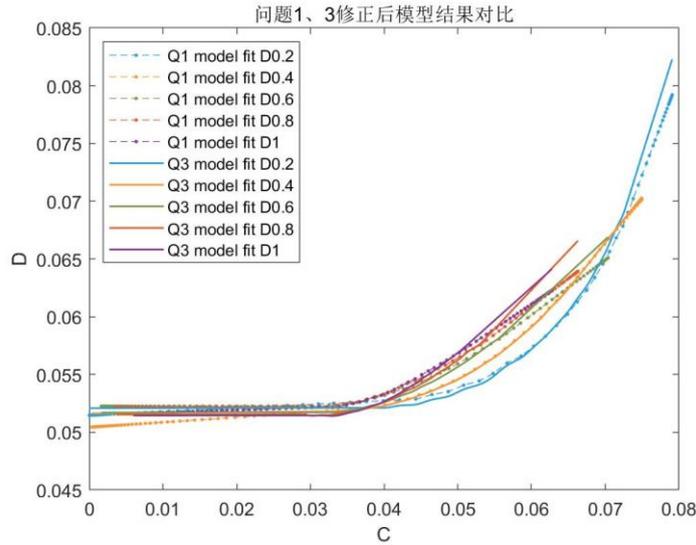


图52 数据量为 1001 的条件下，模型修正后的扩散系数 D 计算结果与问题一得到的扩散系数 D 的对比

根据图 53 可以得出，总体来说本模型在数据量较大时的误差大于数据量较小时的误差，说明本模型在较大数据量时出现了过度拟合的情况，使误差增大。但是趋势并没有表现出误差和数据量的大小呈严格的正相关性。但是模型的数据量并不是越小越好，数据量太少会导致模型无法完成充分的拟合，导致模型对新数据的预测结果差。

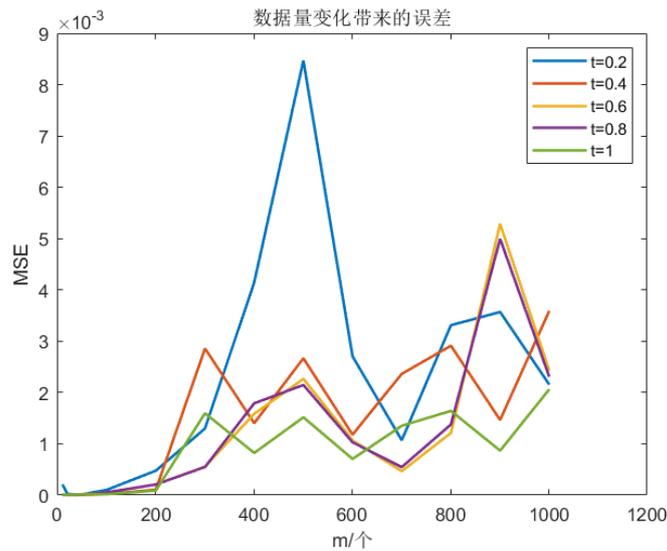


图53 随数据量变化的误差分布对比

5 模型评价和改进

5.1 模型的优点

(1) 在处理问题一时，本文充分结合实测数据与 Fick 定律，使用一维假设，仅考虑单组元体系。在进行数据拟合时，对反正切函数、双曲正切函数以及 sigmoid 函数进行了综合细致的比较，并给出了不同函数的拟合误差，得到了最优的拟合效果，从而保证了模型求解扩散通量的准确性。并将模型预测值与测量值进行比对，发现在两端处存在较大差异后，将模型做了修正，并验证了引入修正项的正确性，这样得到的模型精度高，贴合实际，具有较高的应用价值。

(2) 在处理问题二时，面对组元的扩散通量 J 和浓度 C 的空间分布只测量二者之一的情况下，模型运用 Fick 定律求出未知分布，将复杂的问题转化为较为简单的问题 1，将模型与预测值比对后，合理设置参数进行修正，模型的输出结果符合题目要求，能解决实际问题。

(3) 在处理问题三时，模型对认为产生随机测量误差较小（0.1% ~0.3%）的时候精度评估效果较为良好，但在随机测量误差较大（0.4% ~0.5%）的时候精度评估效果出现较大的误差；同时，模型显示在数据量较小的时候对数据计算的精度较高，而数据量增大的时候误差将会普遍增大。

(4) 通过推广问题一、二和三的算法与模型，使模型可应用与高维和多组分等复杂情形。而实际工程中所涉及的扩散过程基本都属于多组分物系，所以本文所建立的针对问题 4 的模型更具有实际应用价值。

5.2 模型的不足

(1) 本文提出的模型受拟合效果的影响较大，由于时间问题没有对更多的函数进行检验，可能存在拟合效果更好的函数，因而无法保证模型的精度为最优；

(2) 实际问题中，扩散过程可能还会受到温度与压力的影响，但本文未能考虑到这些因素的影响，一定程度上影响了问题四中模型的准确性。

5.3 模型的改进和推广

改进：在建立问题四的模型时，由于不同的温度与压力会影响扩散过程，可以考虑加入温度与压力的影响，从而提高模型的实际应用能力。

推广：本文基于问题一的已知条件，根据所建立的模型进行后续迭代时间的扩散通量 J 和组元浓度 C 在未来不同 t 时刻的发展情况进行模拟，其中，图 54 ~图 63 分别是 $t = 1.2$ 时刻逐渐递增到 $t = 100$ 时刻。可以发现，当 $t = 5$ 时刻扩散通量 J 和组元浓度 C 就以基本认为趋于平稳状态。由于篇幅有限，为方便直观理解，作者制作了该物质扩散过程的动画，详见本文附件支撑材料【cartoon.gif】。

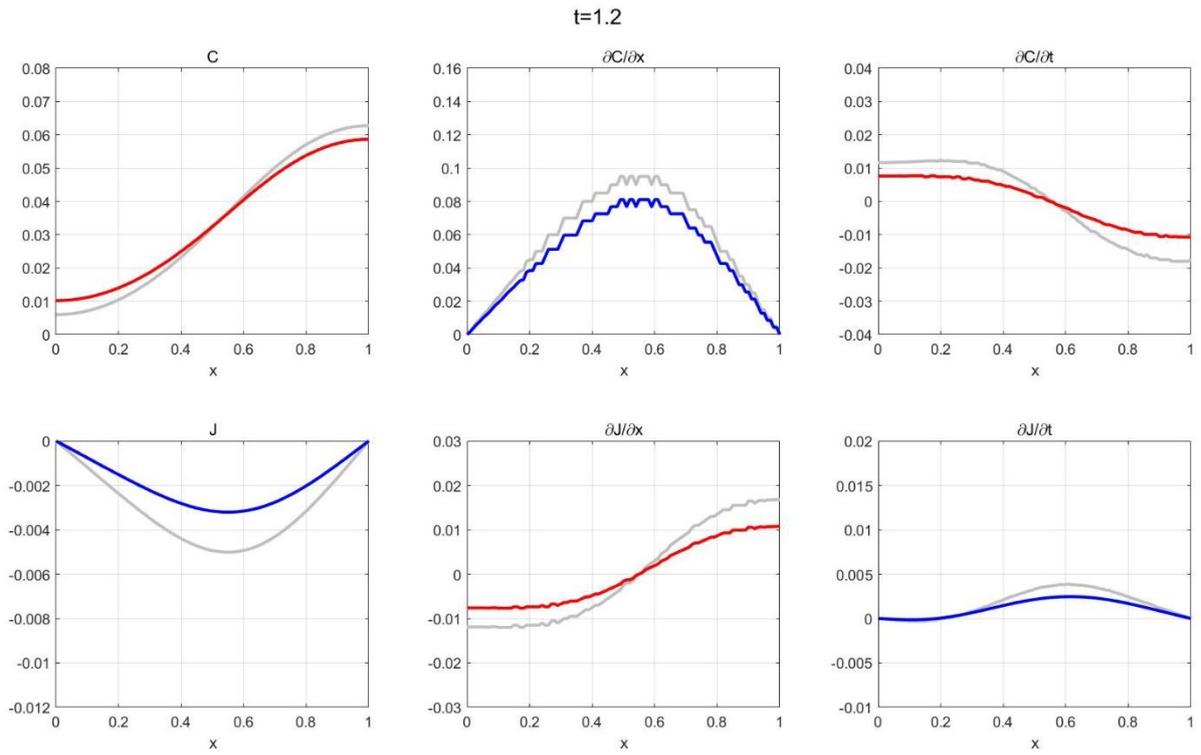


图54 $t=1.2$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

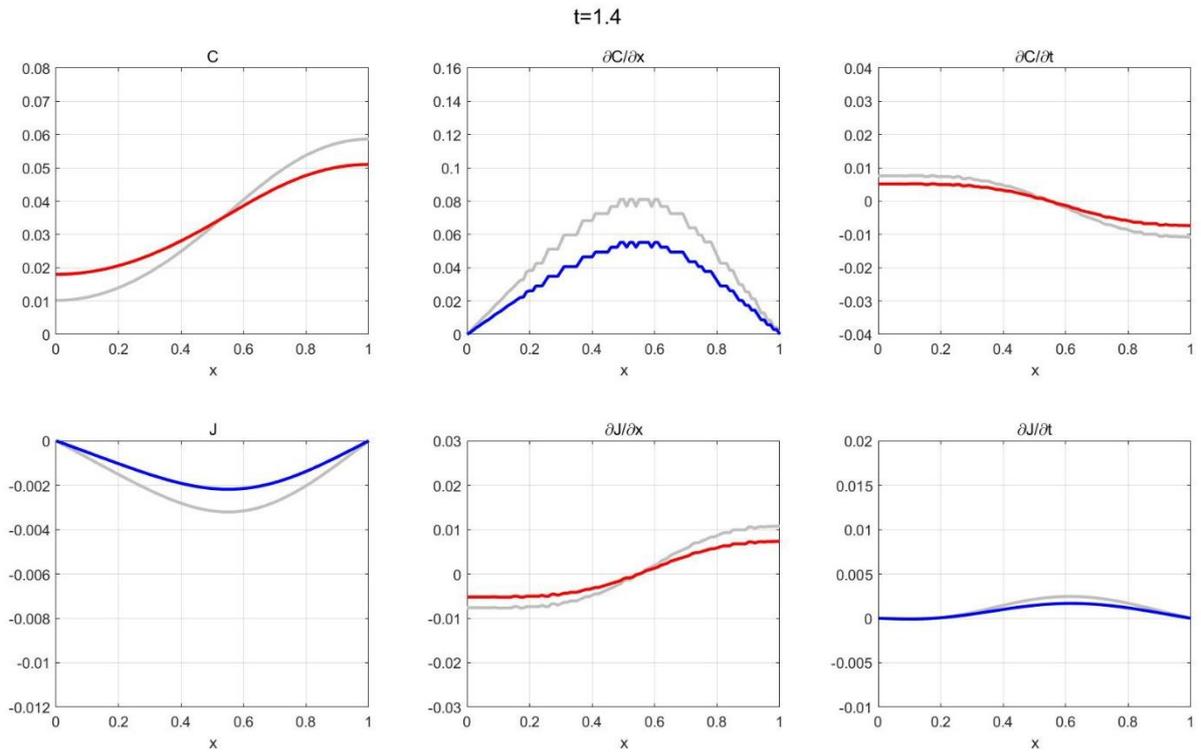


图55 $t=1.4$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

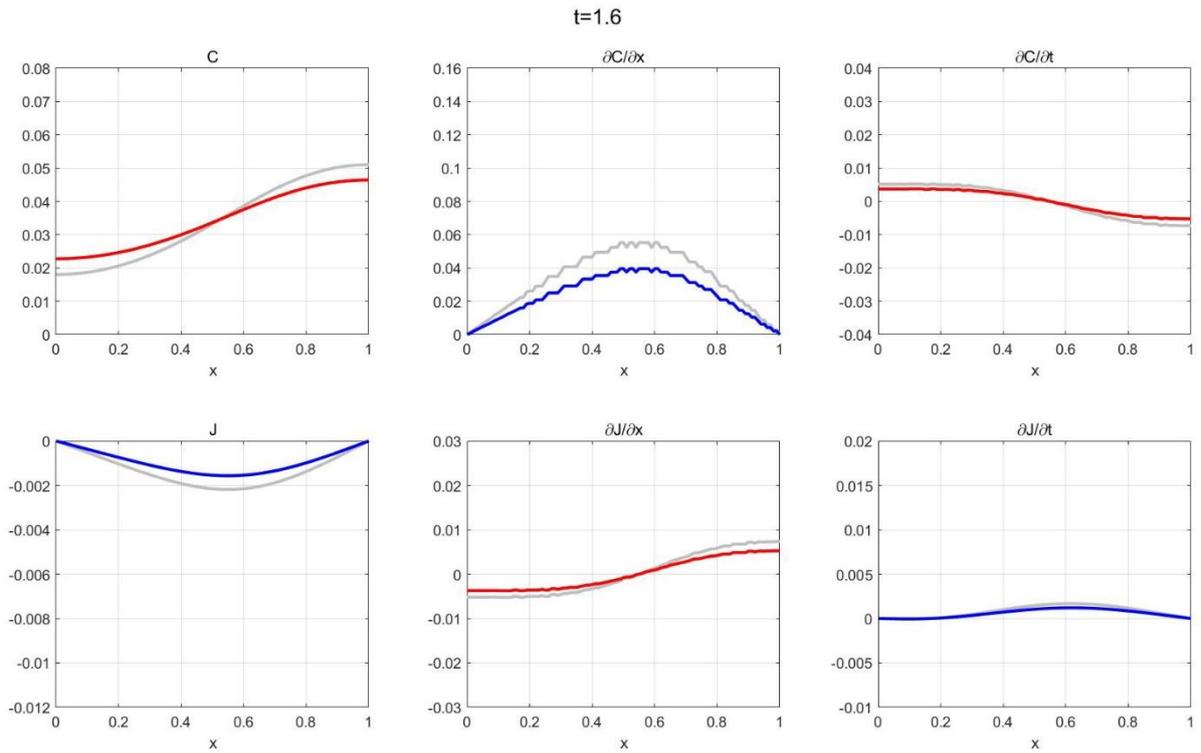


图56 $t=1.6$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

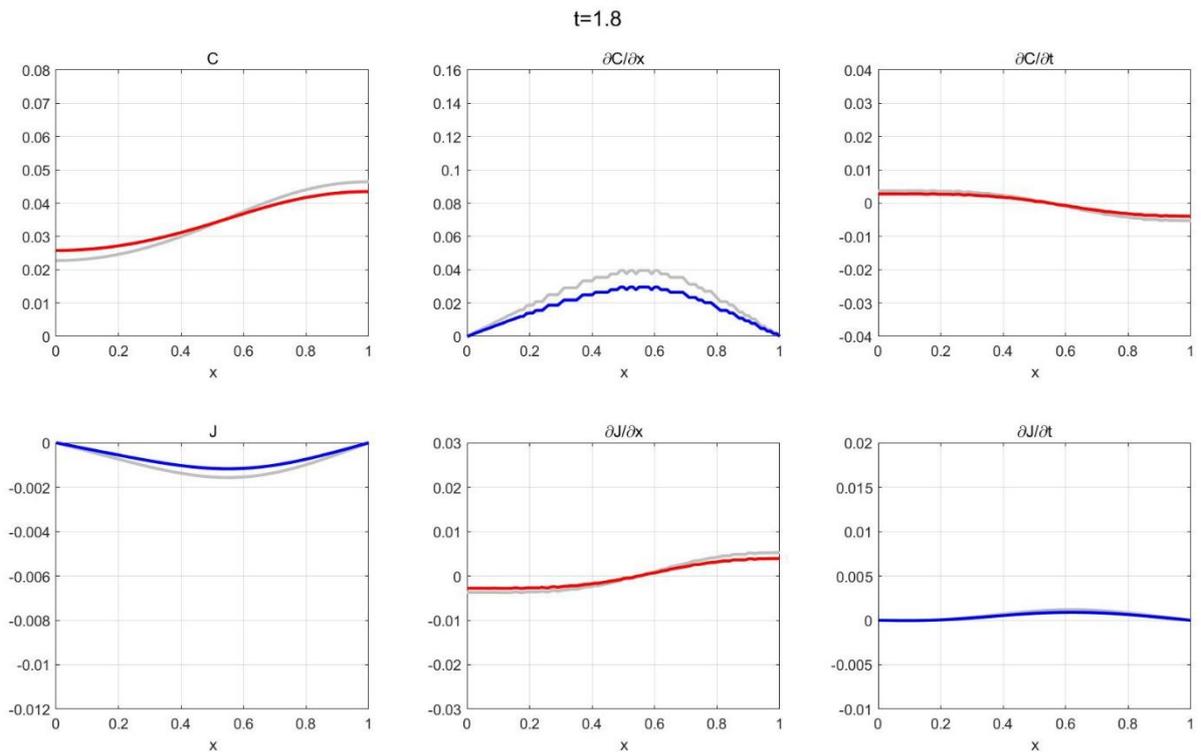


图57 $t=1.8$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

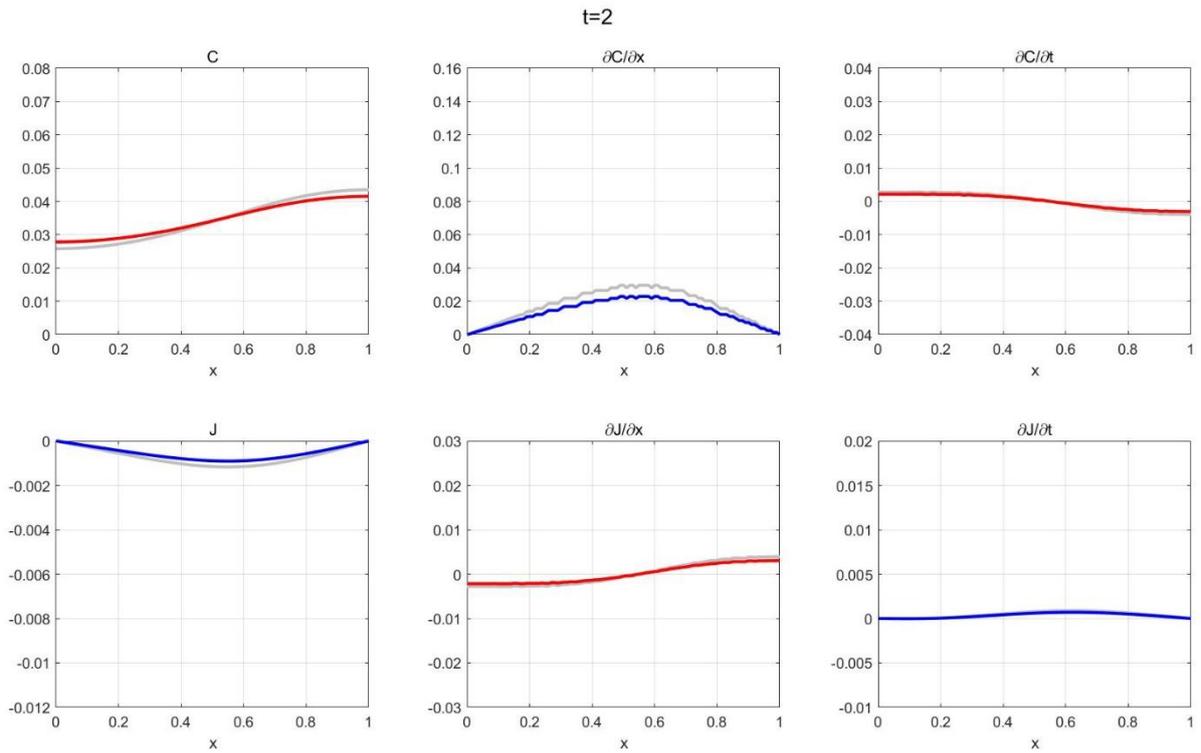


图58 $t=2$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

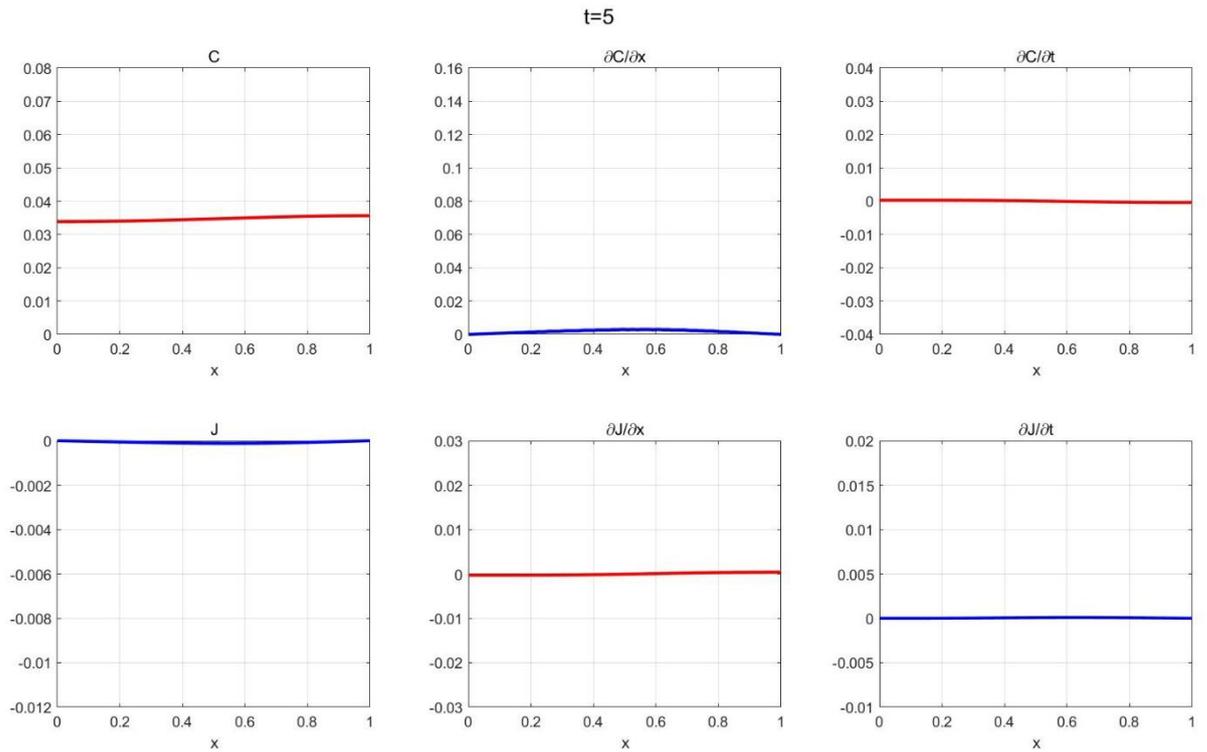


图59 $t=5$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

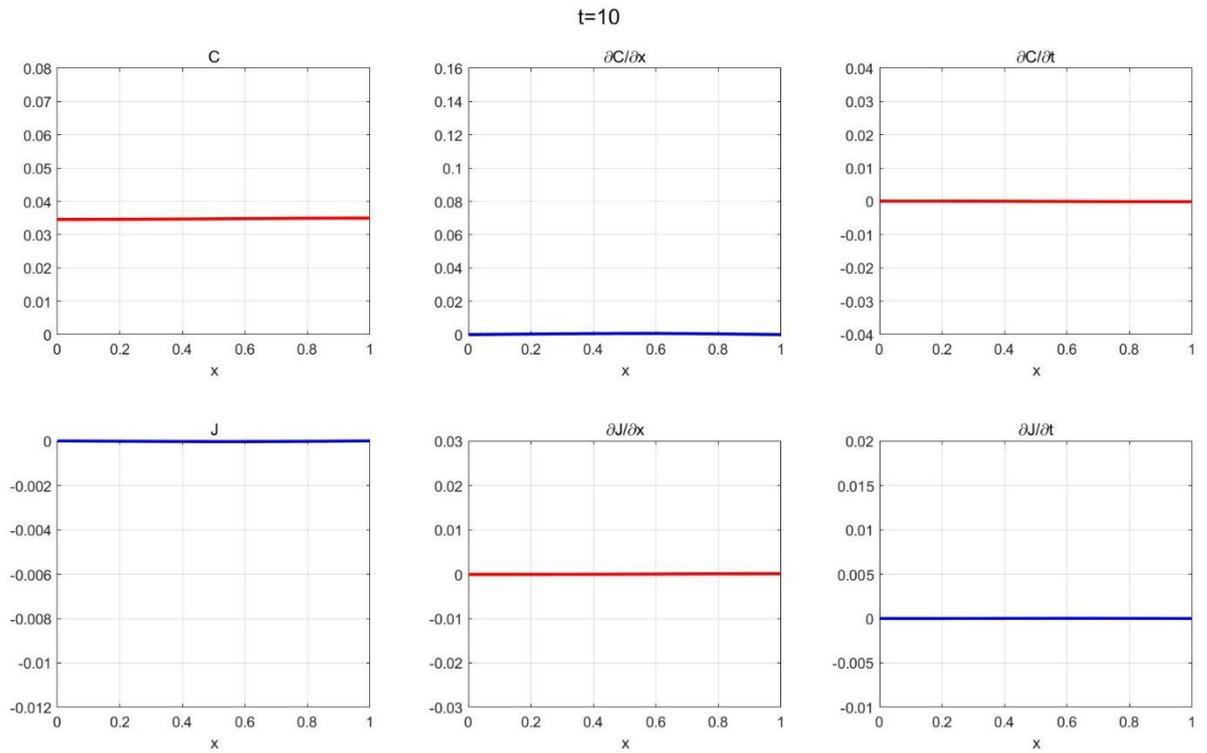


图60 $t=10$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

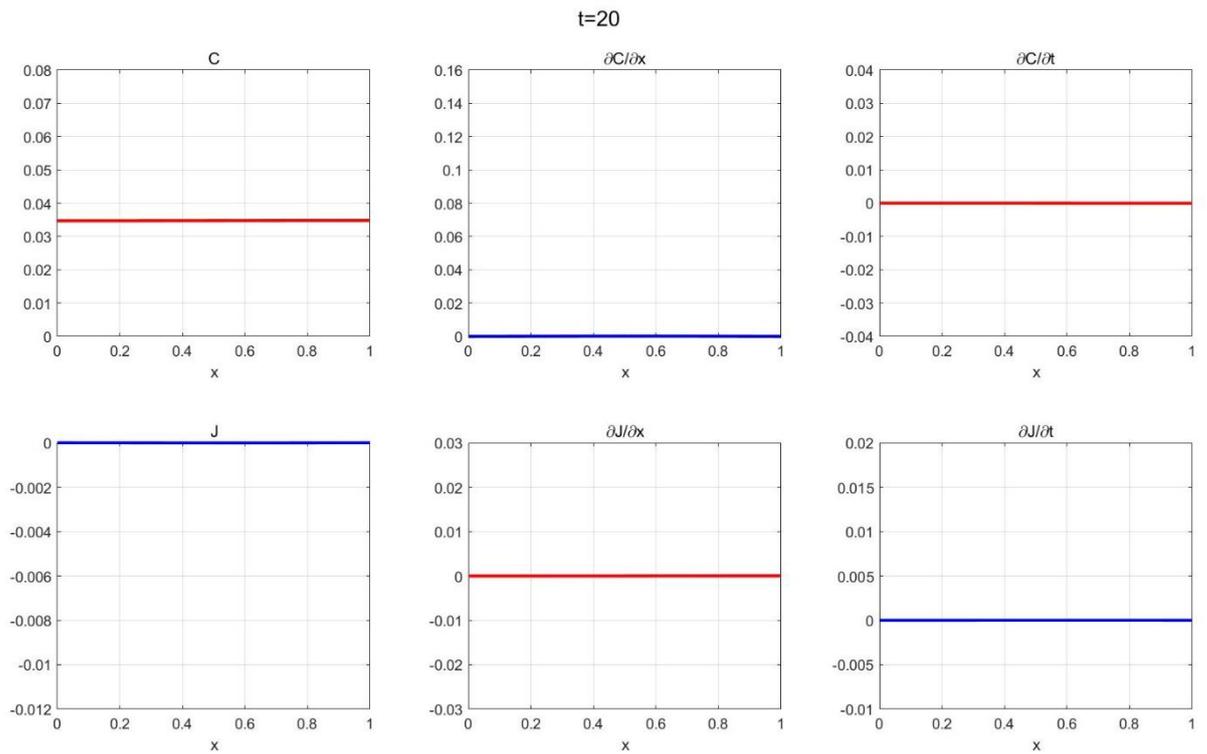


图61 $t=20$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

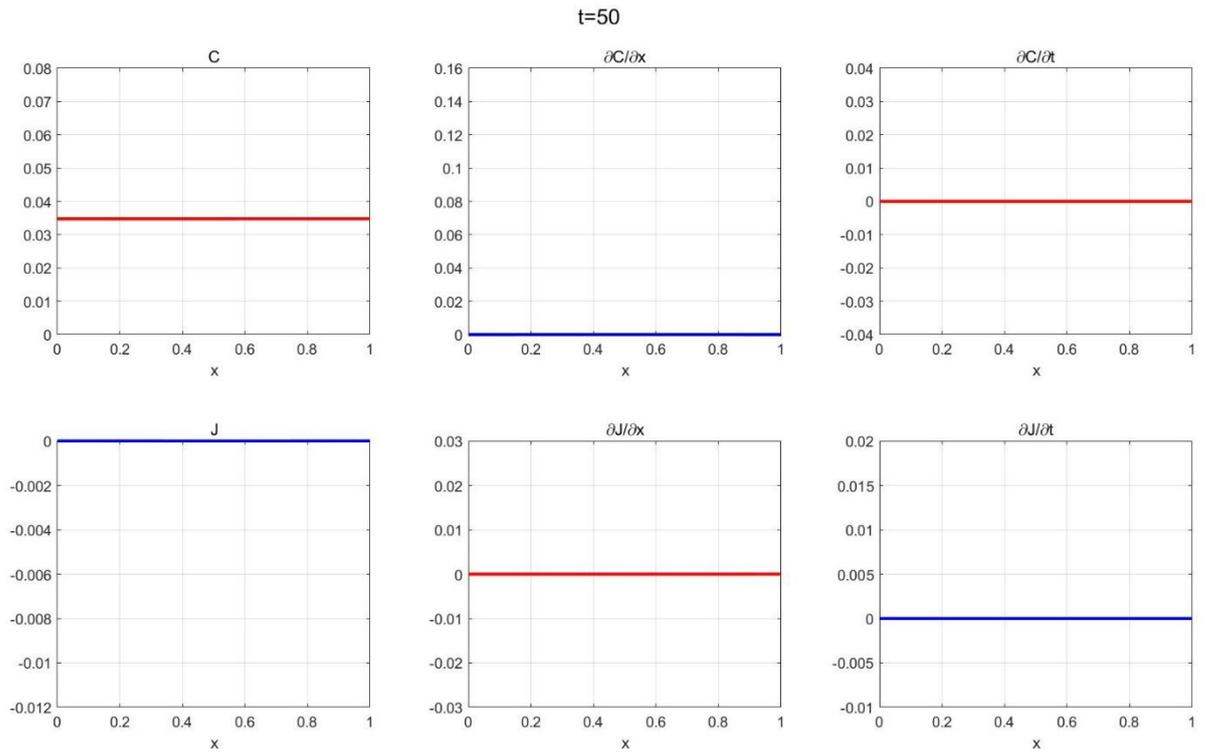


图62 $t=50$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

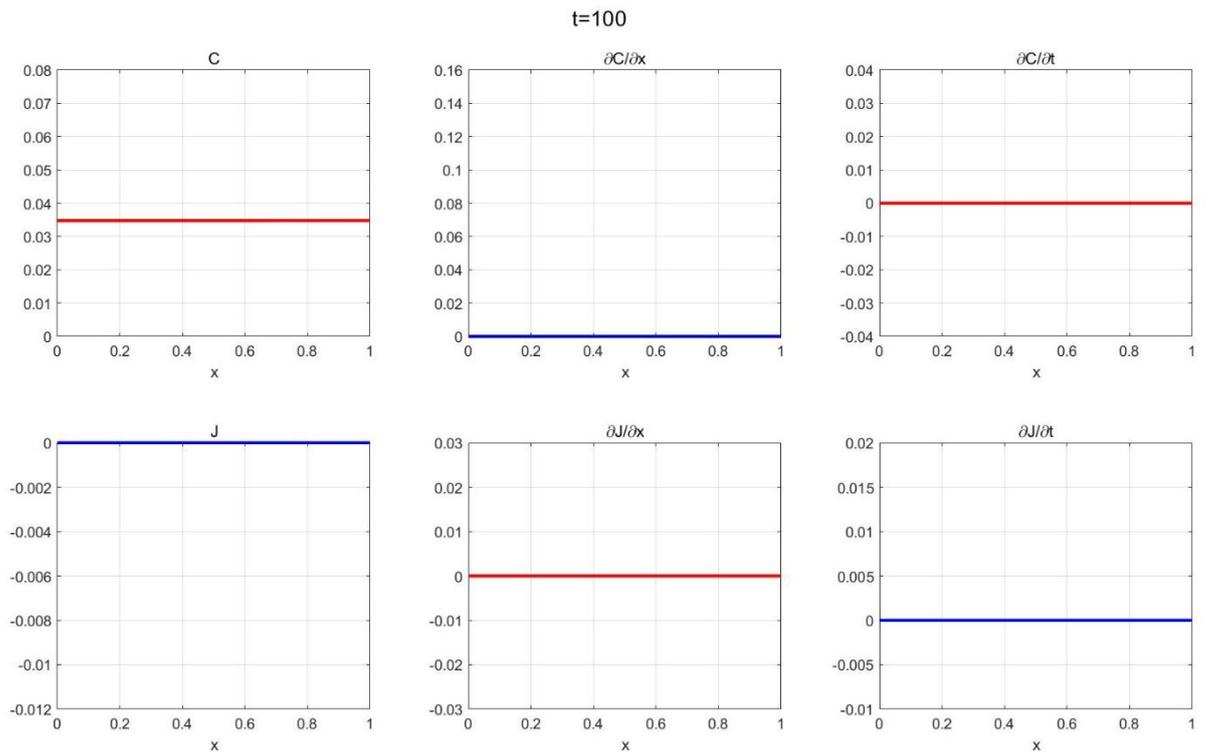


图63 $t=100$ 时刻扩散通量 J 和体积浓度 C 的及其时空偏导数的分布情况

参考文献

- [1] 谢清馨. 基于地层信息计算场地卓越周期的频域方法研究[D].西安理工大学,2021.DOI:10.27398/d.cnki.gxalu.2021.000130.
- [2] 刘一博. 扩散方程反问题求解的 B 样条小波方法[D].哈尔滨工业大学,2021.DOI:10.27061/d.cnki.ghgdu.2021.001150.
- [3] 周焕林,严俊,余波,陈豪龙.识别含热源瞬态热传导问题的热扩散系数[J].应用数学和力学,2018,39(02):160-169.
- [4] 王志美,陈传仁.遗传算法理论及其应用发展[J].内蒙古石油化工,2006(09):44-45.
- [5] Dayananda M A. A Direct Derivation of Fick's Law for Multicomponent Diffusion[J]. Journal of Phase Equilibria and Diffusion, 2022.
- [6] 邹雨,王国建,卢丽,朱怀平,刘光祥,袁玉松,杨海元,金之钧.纳米孔隙中页岩气扩散模拟实验和数学模型分析[J].石油实验地质,2021,43(05):844-854.
- [7] <https://zhuanlan.zhihu.com/p/411798670>
- [8] <https://zhuanlan.zhihu.com/p/29957294>
- [9] <https://zhuanlan.zhihu.com/p/421954743>
- [10] <https://ww2.mathworks.cn/help/matlab/ref/spline.html>

附录

附录 A: 支撑材料列表

表4 支撑材料列表

序号	文件名	材料说明
1	cartoon.gif	物质扩散发展过程动画
2	第 1 题代码*	模型 3.3.1 计算结果与生成图像
3	第 2 题由 C 求 D 代码*	模型 3.3.2 计算结果与生成图像
4	第 2 题由 J 求 D 代码*	模型 3.3.3 3.3.2 计算结果与生成图像
5	第 3 题创造随机误差代码*	模型 3.3.4 3.3.2 计算结果与生成图像
6	第 3 题改变数据量代码*	模型 3.3.4 3.3.2 计算结果与生成图像

附录 B: 主要程序/关键代码

代 码 环 境	<p>操作系统: Windows 10 编程语言: Matlab 编辑器: Matlab R2019a 代码详见: 第 1 题代码\Q1_draw_normalize_fit.m</p> <pre> close all; load('origin_data.mat'); dC_center_0_2 = (C0_2(3:end) - C0_2(1:end-2))/0.02; dC_center_0_4 = (C0_4(3:end) - C0_4(1:end-2))/0.02; dC_center_0_6 = (C0_6(3:end) - C0_6(1:end-2))/0.02; dC_center_0_8 = (C0_8(3:end) - C0_8(1:end-2))/0.02; dC_center_1 = (C1(3:end) - C1(1:end-2))/0.02; D_center_0_2 = - J0_2(2:end-1) ./ dC_center_0_2; D_center_0_4 = - J0_4(2:end-1) ./ dC_center_0_4; D_center_0_6 = - J0_6(2:end-1) ./ dC_center_0_6; D_center_0_8 = - J0_8(2:end-1) ./ dC_center_0_8; D_center_1 = - J1(2:end-1) ./ dC_center_1; D_center_0_2(D_center_0_2==inf)=0; D_center_0_4(D_center_0_4==inf)=0; D_center_0_6(D_center_0_6==inf)=0; D_center_0_8(D_center_0_8==inf)=0; D_center_1(D_center_1==inf)=0; </pre>
------------------	--

```

Const1 = 0.0808; Const21 = 11.199; Const22 = 0.5794;
dC_model_0_2 = Const1 * Const21 * exp(-Const21 * x + Const21 * Const22) ./ ((1 +
exp(-Const21 * x + Const21 * Const22)).^2);
Const1 = 0.079; Const21 = 8.1198; Const22 = 0.5695;
dC_model_0_4 = Const1 * Const21 * exp(-Const21 * x + Const21 * Const22) ./ ((1 +
exp(-Const21 * x + Const21 * Const22)).^2);
Const1 = 0.07623; Const21 = 6.7198; Const22 = 0.5564;
dC_model_0_6 = Const1 * Const21 * exp(-Const21 * x + Const21 * Const22) ./ ((1 +
exp(-Const21 * x + Const21 * Const22)).^2);
Const1 = 0.0714; Const21 = 6.1099; Const22 = 0.543;
dC_model_0_8 = Const1 * Const21 * exp(-Const21 * x + Const21 * Const22) ./ ((1 +
exp(-Const21 * x + Const21 * Const22)).^2);
Const1 = 0.06565; Const21 = 5.8198; Const22 = 0.5314;
dC_model_1 = Const1 * Const21 * exp(-Const21 * x + Const21 * Const22) ./ ((1 + exp(-
Const21 * x + Const21 * Const22)).^2);
D_model_0_2 = - J0_2 ./ dC_model_0_2;
D_model_0_4 = - J0_4 ./ dC_model_0_4;
D_model_0_6 = - J0_6 ./ dC_model_0_6;
D_model_0_8 = - J0_8 ./ dC_model_0_8;
D_model_1 = - J1 ./ dC_model_1;

dD_model_0_2 = (D_model_0_2(2:end)-D_model_0_2(1:end-1))./(C0_2(2:end)-
C0_2(1:end-1));
max_dD_model = 0; min_dD_model = 999;
for i = 1:100
    if i > 50 && dD_model_0_2(i) > max_dD_model
        tail = i;
        max_dD_model = dD_model_0_2(i);
    end
    if i > 40 && i < 60 && abs(dD_model_0_2(i)) < min_dD_model
        head = i;
        min_dD_model = abs(dD_model_0_2(i));
    end
end
D_model_fit_0_2 = D_model_0_2;
D_model_fit_0_2(1:head) = D_model_0_2(head) + min_dD_model * (C0_2(1:head) -
C0_2(head));
D_model_fit_0_2(tail:end) = D_model_0_2(tail) + max_dD_model * (C0_2(tail:end) -
C0_2(tail));

dD_model_0_4 = (D_model_0_4(2:end)-D_model_0_4(1:end-1))./(C0_4(2:end)-
C0_4(1:end-1));
max_dD_model = 0; min_dD_model = 999;
for i = 1:100
    if i > 50 && dD_model_0_4(i) > max_dD_model
        tail = i;
        max_dD_model = dD_model_0_4(i);
    end
    if i > 40 && i < 60 && abs(dD_model_0_4(i)) < min_dD_model
        head = i;

```

```

    min_dD_model = abs(dD_model_0_4(i));
end
end
D_model_fit_0_4 = D_model_0_4;
D_model_fit_0_4(1:head) = D_model_0_4(head) + min_dD_model * (C0_4(1:head) -
C0_4(head));
D_model_fit_0_4(tail:end) = D_model_0_4(tail) + max_dD_model * (C0_4(tail:end) -
C0_4(tail));

dD_model_0_6 = (D_model_0_6(2:end)-D_model_0_6(1:end-1))./(C0_6(2:end)-
C0_6(1:end-1));
max_dD_model = 0; min_dD_model = 999;
for i = 1:100
    if i > 50 && dD_model_0_6(i) > max_dD_model
        tail = i;
        max_dD_model = dD_model_0_6(i);
    end
    if i > 40 && i < 60 && abs(dD_model_0_6(i)) < min_dD_model
        head = i;
        min_dD_model = abs(dD_model_0_6(i));
    end
end
end
D_model_fit_0_6 = D_model_0_6;
D_model_fit_0_6(1:head) = D_model_0_6(head) + min_dD_model * (C0_6(1:head) -
C0_6(head));
D_model_fit_0_6(tail:end) = D_model_0_6(tail) + max_dD_model * (C0_6(tail:end) -
C0_6(tail));

dD_model_0_8 = (D_model_0_8(2:end)-D_model_0_8(1:end-1))./(C0_8(2:end)-
C0_8(1:end-1));
max_dD_model = 0; min_dD_model = 999;
for i = 1:100
    if i > 50 && dD_model_0_8(i) > max_dD_model
        tail = i;
        max_dD_model = dD_model_0_8(i);
    end
    if i > 40 && i < 60 && abs(dD_model_0_8(i)) < min_dD_model
        head = i;
        min_dD_model = abs(dD_model_0_8(i));
    end
end
end
D_model_fit_0_8 = D_model_0_8;
D_model_fit_0_8(1:head) = D_model_0_8(head) + min_dD_model * (C0_8(1:head) -
C0_8(head));
D_model_fit_0_8(tail:end) = D_model_0_8(tail) + max_dD_model * (C0_8(tail:end) -
C0_8(tail));

dD_model_1 = (D_model_1(2:end)-D_model_1(1:end-1))./(C1(2:end)-C1(1:end-1));
max_dD_model = 0; min_dD_model = 999;
for i = 1:100

```

```

if i > 50 && dD_model_1(i) > max_dD_model
    tail = i;
    max_dD_model = dD_model_1(i);
end
if i > 40 && i < 60 && abs(dD_model_1(i)) < min_dD_model
    head = i;
    min_dD_model = abs(dD_model_1(i));
end
end
D_model_fit_1 = D_model_1;
D_model_fit_1(1:head) = D_model_1(head) + min_dD_model * (C1(1:head) - C1(head));
D_model_fit_1(tail:end) = D_model_1(tail) + max_dD_model * (C1(tail:end) - C1(tail));

figure;
plot((C0_2(2:end-1)-min(C0_2(2:end-1)))/(max(C0_2(2:end-1))-min(C0_2(2:end-1))),D_center_0_2,'--','color',[40 160 220]/255);hold on;
plot((C0_4(2:end-1)-min(C0_4(2:end-1)))/(max(C0_4(2:end-1))-min(C0_4(2:end-1))),D_center_0_4,'--','color',[250 150 50]/255);hold on;
plot((C0_6(2:end-1)-min(C0_6(2:end-1)))/(max(C0_6(2:end-1))-min(C0_6(2:end-1))),D_center_0_6,'--','color',[125 150 75]/255);hold on;
plot((C0_8(2:end-1)-min(C0_8(2:end-1)))/(max(C0_8(2:end-1))-min(C0_8(2:end-1))),D_center_0_8,'--','color',[225 90 40]/255);hold on;
plot((C1(2:end-1)-min(C1(2:end-1)))/(max(C1(2:end-1))-min(C1(2:end-1))),D_center_1,'--','color',[135 55 145]/255);hold on;

plot((C0_2(2:end-1)-min(C0_2(2:end-1)))/(max(C0_2(2:end-1))-min(C0_2(2:end-1))),D_model_fit_0_2(2:end-1),'-','color',[40 160 220]/255,'linewidth',1);hold on;
plot((C0_4(2:end-1)-min(C0_4(2:end-1)))/(max(C0_4(2:end-1))-min(C0_4(2:end-1))),D_model_fit_0_4(2:end-1),'-','color',[250 150 50]/255,'linewidth',1);hold on;
plot((C0_6(2:end-1)-min(C0_6(2:end-1)))/(max(C0_6(2:end-1))-min(C0_6(2:end-1))),D_model_fit_0_6(2:end-1),'-','color',[125 150 75]/255,'linewidth',1);hold on;
plot((C0_8(2:end-1)-min(C0_8(2:end-1)))/(max(C0_8(2:end-1))-min(C0_8(2:end-1))),D_model_fit_0_8(2:end-1),'-','color',[225 90 40]/255,'linewidth',1);hold on;
plot((C1(2:end-1)-min(C1(2:end-1)))/(max(C1(2:end-1))-min(C1(2:end-1))),D_model_fit_1(2:end-1),'-','color',[135 55 145]/255,'linewidth',1);hold on;

ylim([0.045 0.085]);
xlabel('normal(C)');
ylabel('D');
legend('data D0.2','data D0.4','data D0.6','data D0.8','data D1','model fit D0.2','model fit D0.4','model fit D0.6','model fit D0.8','model fit D1','location','northwest');
title('中心差、修正后模型归一化结果对比');

figure;
cut = 0;
average_Error0_2 = mean((D_model_fit_0_2(2+cut:end-1-cut)-D_center_0_2(1+cut:end-cut)).^2);
average_Error0_4 = mean((D_model_fit_0_4(2+cut:end-1-cut)-D_center_0_4(1+cut:end-cut)).^2);
average_Error0_6 = mean((D_model_fit_0_6(2+cut:end-1-cut)-D_center_0_6(1+cut:end-

```

```

cut).^2);
average_Error0_8 = mean((D_model_fit_0_8(2+cut:end-1-cut)-D_center_0_8(1+cut:end-
cut)).^2);
average_Error1 = mean((D_model_fit_1(2+cut:end-1-cut)-D_center_1(1+cut:end-
cut)).^2);
average_Error = [average_Error0_2 average_Error0_4 average_Error0_6 average_Er-
ror0_8 average_Error1]

plot((C0_2(2+cut:end-1-cut)-min(C0_2(2+cut:end-1-cut)))/(max(C0_2(2+cut:end-1-cut))-
min(C0_2(2+cut:end-1-cut))),D_model_fit_0_2(2+cut:end-1-cut)-D_cen-
ter_0_2(1+cut:end-cut)).^2,'-', 'color',[40 160 220]/255);hold on;
plot((C0_4(2+cut:end-1-cut)-min(C0_4(2+cut:end-1-cut)))/(max(C0_4(2+cut:end-1-cut))-
min(C0_4(2+cut:end-1-cut))),D_model_fit_0_4(2+cut:end-1-cut)-D_cen-
ter_0_4(1+cut:end-cut)).^2,'-', 'color',[250 150 50]/255);hold on;
plot((C0_6(2+cut:end-1-cut)-min(C0_6(2+cut:end-1-cut)))/(max(C0_6(2+cut:end-1-cut))-
min(C0_6(2+cut:end-1-cut))),D_model_fit_0_6(2+cut:end-1-cut)-D_cen-
ter_0_6(1+cut:end-cut)).^2,'-', 'color',[125 150 75]/255);hold on;
plot((C0_8(2+cut:end-1-cut)-min(C0_8(2+cut:end-1-cut)))/(max(C0_8(2+cut:end-1-cut))-
min(C0_8(2+cut:end-1-cut))),D_model_fit_0_8(2+cut:end-1-cut)-D_cen-
ter_0_8(1+cut:end-cut)).^2,'-', 'color',[225 90 40]/255);hold on;
plot((C1(2+cut:end-1-cut)-min(C1(2+cut:end-1-cut)))/(max(C1(2+cut:end-1-cut))-
min(C1(2+cut:end-1-cut))),D_model_fit_1(2+cut:end-1-cut)-D_center_1(1+cut:end-
cut)).^2,'-', 'color',[135 55 145]/255);hold on;

ylim([-2e-4 1e-3]);
xlabel('normal(C)');
ylabel('error(D)');
legend('D0.2','D0.4','D0.6','D0.8','D1','location','northwest');
title('修正后模型相对于中心差归一化结果误差');

figure;
plot(C0_2(2:end-1),D_model_fit_0_2(2:end-1),'-', 'color',[40 160 220]/255,'lin-
ewidth',1);hold on;
plot(C0_4(2:end-1),D_model_fit_0_4(2:end-1),'-', 'color',[250 150 50]/255,'lin-
ewidth',1);hold on;
plot(C0_6(2:end-1),D_model_fit_0_6(2:end-1),'-', 'color',[125 150 75]/255,'lin-
ewidth',1);hold on;
plot(C0_8(2:end-1),D_model_fit_0_8(2:end-1),'-', 'color',[225 90 40]/255,'lin-
ewidth',1);hold on;
plot(C1(2:end-1),D_model_fit_1(2:end-1),'-', 'color',[135 55 145]/255,'linewidth',1);hold
on;

ylim([0.02 0.1]);
xlabel('C');
ylabel('D');
legend('model fit D0.2','model fit D0.4','model fit D0.6','model fit D0.8','model fit D1','lo-
cation','northwest');
title('修正后模型结果');

```

```
save('Q1_Ans','D_center_0_2','D_center_0_4','D_center_0_6','D_center_0_8','D_center_1',  
'D_model_fit_0_2','D_model_fit_0_4','D_model_fit_0_6','D_model_fit_0_8','D_model_fit_1',  
'D_model_0_2','D_model_0_4','D_model_0_6','D_model_0_8','D_model_1');
```