ADIOSS — 系统仿真自动诊断

Di Jiang[1,2], Sebastian Rodriguez[1],Herv´e Colin[2], Yves Tourbier[1,2], 和 Francisco Chinesta[11] PIMM，巴黎国立高等工艺学院，法国巴黎 [2] 雷诺集团，法国盖扬库尔

arXiv:2603.13504v1 [cs.CE] 13 Mar 2026

摘要 – 汽车工程在整个设计过程中广泛使用数值仿真。数值模型的开发、通过实验测试进行验证，以及在车辆和发动机项目期间对其进行更新，构成了核心工程活动。然而，这项活动必须持续演进，以降低成本和缩短交付周期。

在此背景下，我们提出了一种方法，用于在系统级仿真工作流中检测故障模块。该工作流以 0D 模型构成的图表示，并在模型更新后进行检测。所提出的方法只需极少量的系统仿真，因此可以轻松集成到现有工程流程中。该方法设计为一个工具箱，基于成熟且经过广泛验证的技术，包括 *Dynamic Mode Decomposition*——常用于 $3D$ 模型降阶的技术、线性规划和自编码器。

关键词。0D车辆仿真，模块化架构，DMDc，数据驱动系统辨识，模块交互分析，汽车仿真平台

目录

1 引言 2 1.1 使用的符号.3 2 模块化验证与敏感性分析 4 2.1 一次验证一个模块.4 2.2 敏感性分析.4 3 案例研究 5 3.1 工作流描述.5 3.2 模块版本变更的表示.5 3.3 关键参数的变化程度.6 3.4 指定变量.6 4 带参考的单次仿真方法（MixED-DMD）6 4.1 使用 DMDc 进行参考模型辨识.8 4.2 不使用潜在嵌入的 MixED-DMD 模块检测.10 4.3 使用潜在嵌入的 MixED-DMD 模块检测.13 5 带参考的双仿真方法（NoDyn）15 6 结论 17

1. 引言

数值仿真在汽车设计流程中发挥着核心作用。用于决策的模型通常分为两类：一类是 $3D$ 基于有限元法的模型 [1], ，当设计或计算机辅助设计（CAD）模型可用时采用；另一类是 $0D$ 模型，其依赖于对物理过程进行受控简化 [2], [3],[4]。本文特别关注 $0D$ 模型，这类模型也称为“系统模型”。为避免与统计模型混淆，本文将其统称为“模块”。

$0D$ 模型或模块广泛用于在各种客户驾驶循环中快速评估车辆性能。这些模块根据函数 $x_{t+1} = f(x_t)$ 动态仿真状态 x 的变化；在复杂的计算链中，当多个模块通过其输入/输出变量连接起来时，这些计算链被称为工作流。西门子为雷诺使用的仿真平台 [5] 阐释了这一原理。

工作流由不同的工程领域提供，每个领域都会提供经过更新且单独验证的模块。例如，用于仿真电动汽车在给定路线上的行为的工作流，可能包括电池、电机、逆变器、电池冷却系统、空调系统，以及预测能耗和续航里程时所有消耗车载能源的组件。标准工作流包含十多个模块。模块图由系统仿真平台执行，该平台通过所谓的弱耦合[6]将这些模块集成起来。

我们的研究重点是系统模块的验证流程。在验证周期非常短且仿真预算有限的汽车工业环境中，确保模块从设计阶段起就具备可靠性至关重要。因此，通常在项目末期开展的实验验证，自建模阶段起就要求具备较高的质量水平。具体背景如下：

- 有一套在客户驾驶循环上经过验证的参考仿真工作流，记为 W_0。例如，这套工作流及其 m 模块 $M_0^1 ... M_0^m$曾在上一个车辆项目结束时完成验证；其结果与最终车辆验证测试进行了比较，并被认为令人满意；
- 各工程团队和外部供应商交付了新版本 W_1（$M_1^1 ... M_1^m$）；求解器甚至平台本身也可能已更新。对于客户驾驶循环，这套新工作流得出的结果与旧版本不同，例如在能耗、续航里程、热舒适性等全局指标方面。由于先前的结果作为参考，在未来车辆项目中使用新版本之前，必须解释这些差异。本文后续将导致这些差异的模块称为“故障模块”。

工作流的两个版本之间可能会发生许多变化，而这些变化对性能指标的影响并不总是显而易见。此外，外部供应商不会详细说明所有已实施的变更，也不会提供其源代码的访问权限。由于工作流用于正在进行的项目，因此验证与纠错流程必须具备高度敏捷性。必须能够快速发现差异的来源，以便及时通知相关工程团队或供应商，并确保项目团队始终使用经过验证的版本。问题还在于，模块的最终用户不一定是数值仿真专家。专家开发的模块往往会在其他情境中重复使用，有时使用者并未深入了解其局限性。这种情况凸显了对自动化工具的需求：它能够以**最低成本并在短时间内**高效识别复杂仿真工作流中的故障模块。

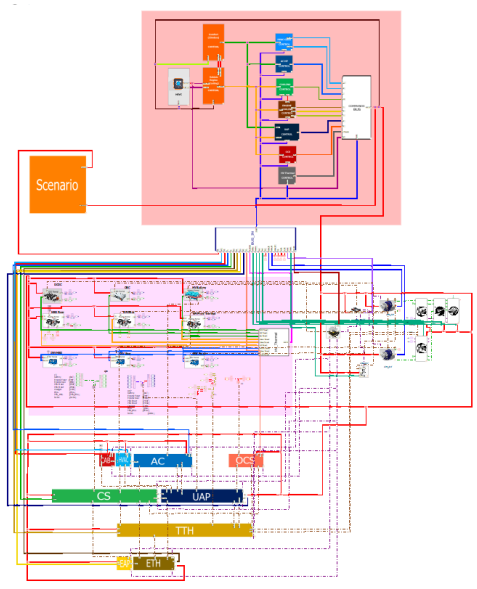


图1：仿真工作流示例。每个方框代表一个模块，可使用$Amesim$、$Simulink$、$Matlab$或基于映射的模型实现。此处使用平台形式化方法表示模块图。连线表示输入/输出变量、能量流、控制信号等。

1.1. 所用符号

系统仿真由 在每个时间步执行函数图。函数**称为模块** ，以性质各异的变量作为输入 不同性质：

- 其值由方程控制的物理变量。在本文中，这些变量必须使用统计方法进行建模；
- 应用于工作流中计算出的物理量、并用于控制目的的阈值，其中控制本身通过模块实现。例如，自适应巡航控制在超过 $30\,km/h$ 时激活；
- 控制变量或定义客户驾驶循环的变量，其值随时间设定，因此无需建模。例如，道路坡度、外部温度、驾驶循环的速度设定值、等。

本文其余部分使用以下符号：

- 时间步记为 t，其取值范围为 1 至 n；
- 参考模块记为 M_0^j，其中 j的取值范围为 1 至 m；
- 模块 M_1^j是 M_0^j的新版本。其目的是确定它是否存在故障，也就是说，是否与 M_0^j确实不同；
- 模块 M^j的变量记作 $x_{j,\,i}$（或 $x_{t,\,j,\,i}$在明确考虑时间索引时）。如果不区分模块结构，则整个工作流的变量统一记作 x_i，其中 i 的取值范围为 1 到 p，涵盖所有物理变量、控制变量和驾驶循环定义变量；
- 模块内部的变量集合由时间步 t时的状态向量 x_t 表示。在仿真过程中，每个时间步都会在各模块的输入端或输出端收集每个变量 $x_{t,\,i}$ （或 $x_{t,\,j,\,i}$）的值；
- 仿真会生成一个记作 $T_{t,\,i}$的数据表，其中时间按行排列（在参考文献中引用的出版物里，通常按列排列）；
- T_0 表示由参考仿真 W_0生成的数据表；
- “试验设计”表示针对布尔变量的试验设计。 DoE 表示将所有 0 和 1 值互换后的同一设计。

由于仿真平台复杂，且需要获得准确结果，计算时间远超实时运行时间。例如，重现一个持续 30 分钟、采样时间为 0.5 秒的客户驾驶循环，可能需要十多个小时。工作流验证过程涉及许多周期，这些周期代表着差异很大的情况或法规要求。因此，所提出的方法必须尽可能减少仿真次数，以便能够系统地使用。

2. 模块化验证与敏感性分析

2.1. 一次验证一个模块

首先想到的一种方法是逐个验证模块。只需运行仿真，并确保每次只有一个模块与参考配置不同。令 T_0 表示从参考仿真中获得的系统输出轨迹， T_j 表示将模块 M^j替换为更新版本、而所有其他模块均与参考配置保持一致时获得的轨迹。

这种方法需要 $m+1$ 次仿真：一次参考仿真，以及每个模块各进行一次仿真。该方法易于实现，而且可以并行运行仿真以缩短周转时间。通过计算差异 $T_0 - T_j$，可以检测替换模块 M^j 所产生的哪怕非常微小的变化。这使得我们能够判断是否确实引入了变化。

然而，模块之间的交互作用很强。只要修改某个模块，就总能观察到差异，但无法按重要性从高到低对模块变更进行排序。此外，检测能力取决于所仿真的驾驶循环。若在仿真过程中未达到某个已修改的阈值，就无法检测到该变化（当然，任何方法都如此）。也可能出现相反的情况：微小的修改可能产生显著影响，因为另一个模块的更新并未抵消这一影响。大多数情况下，模块虽已更新，但产生的差异很小；真正重要的是它们与其他模块之间的交互作用。第一种方法能够检测较大的变化和高度敏感的模块。需要注意的是，系统仿真工作流只包含敏感模块；没有影响的模块不会被集成到工作流中。

2.2. 敏感性分析

首先开发了一种算法，用于基于试验设计方法检测差异的来源 [7], ，与大多数依赖计算成本高昂的仿真的算法一样。

故障模块的检测可以表述为针对布尔变量的组合搜索。设 X_j 为一个布尔变量：当模块 M_0^j在 W_0中被 M_1^j替换时，该变量等于 1；否则等于 0。类似地，设 Z_j为一个布尔变量：当 M_1^j在 W_1中被 M_0^j替换时，该变量等于 1。我们假设差异$W_1 - W_0$完全由模块造成。目标是找到最能解释 $W_1 - W_0$的向量 X 和 Z。搜索可以仅使用 X、仅使用 Z，或同时使用两者。如果 $X = Z$，但两次仿真的结果不同，那么这些模块并不能完全解释差异 $W_1 - W_0$。此时必须调查纯数值方面的原因，例如替换平台本身（这也可以用一个布尔变量表示），或更改用于特定模块的求解器。

最简单的算法仅使用 X，相当于应用试验设计来搜索最优解：

1. 为每个模块分配一个布尔变量；2. 选择一个 DoE。3. 运行参考仿真 W_0；4. 运行由 DoE定义的 N 仿真；5. 计算试验设计中每项试验所要解释的全局性能指标，例如电动汽车的预测续航里程。该指标记为 $W_0(X_i)$；6. 构造响应向量， $\forall t \in 1..N, \quad Y_t = W_0(X_t) - W_0$；ˆ 7. 建立回归模型 $Y_t = f(X_t)$，以模块替换为自变量解释相对于参考值的偏差。模型构建包括变量选择；8. 使用回归模型确定使差异 $W_1 - W_0$最小的最小 X_{opt}。最小的X_{opt} 会使值为 1 的布尔变量数量最少；9. 运行仿真 X_{opt} 并计算 Y_{opt}，以验证该解；ˆ 10. 如果 Y_{opt} 与 Y_{opt}不同，则说明统计模型中缺少某些项。必须添加额外试验，返回步骤 4，并选择更复杂的统计模型（包括额外交互项）。

寻找最佳回归模型和 X_{opt} 可以结合问题的具体结构来进行。由于变量 X_j 是布尔型的，也就是说是定性的，因此插值没有意义。可能的回归模型集合具有以下形式

$$Y = \alpha_0 + \sum_j \alpha_j.X_j + \sum_{j_1,j_2} \alpha_{j_1,j_2}.X_{j_1}.X_{j_2} + ...\alpha_{j_1..j_m}.\prod_{j=1}^{P} X_j$$

非零系数的最大数量 $\alpha_{j_1...}$ 为 2^m，前提是工作流包含 m个模块。**该模型必然是一个多元线性回归模型**。无需使用任何其他回归方法。

例如，如果只有模块 M_1 和 M_2 会导致差异 $W_1 - W_0$，那么回归模型
可简化为

$$Y = \alpha_0 + \alpha_1.X_1 + \alpha_2.X_2 + \alpha_{1,2}.X_1.X_2$$

备注：如果对相对于参考值的偏差进行建模，并采用全因子设计，那么 α_0 等于零，因为参考配置本身就是全因子设计中的一次试验。

当 $\alpha_{1,2}$ 为零，但要解释差异又必须同时涉及两个模块时，这种情况极不可能发生。这意味着仿真期间，工作流中的两个模块之间没有任何共享变量，相当于两个独立工作流并行运行。这种情况在实践中不会发生；交互项 $X_1.X_2$ 是必不可少的。这一性质会带来重大影响：若要从 m个模块中检测出 P 个故障模块，要想完美解释差异 $W_1 - W_0$，待估计的回归模型就必须包含 2^m 个可能系数中的 2^P 个，这意味着需要进行 2^m 次仿真。在这种情况下，**只有全因子设计才能保证检测出所有故障模块**，但其理论计算成本高得令人无法接受。

在实践中，故障模块通常是已知或疑似的，而且不会一次性修改整个工作流。此外，最初的目标是检测，而非进行详细解释。如果故障模块的数量 P 是固定的，但不知道具体是哪些模块，那么实验设计必须包含从 m个变量中选取 P 个变量的所有全因子设计，这仍然非常昂贵。

寻找最佳回归模型的问题与CTD (组合测试设计， [8])方法论以及自动驾驶车辆和高级驾驶辅助系统（ADAS）中的故障场景搜索有相似之处。人们已开发出许多方法 [9]，但介绍这些方法超出了本文的讨论范围。相关研究最早可追溯到20世纪90年代，至今仍在持续。应用这些方法导致仿真成本高昂，也促使我们开展本项工作。

然而，我们的问题不同于经典的软件测试。在CTD中，各项试验彼此独立，目标是找出会触发错误的试验。对于我们的问题，试验是动态仿真，其中输入和输出会随时间相互作用。我们的问题更接近软件验证，即验证一系列操作（见本文后文讨论的控制变量）。

3. 案例研究

3.1. 工作流描述

所选测试用例用于 评估所提出的方法论，该方法基于由.....开发的开源模型 MathWorks [10], representin g 电动汽车的推进系统。

工作流（见图2）采用模块化方式组织，并由给定的驾驶循环驱动。该循环定义了车辆目标速度随时间的演变（见图3b）。

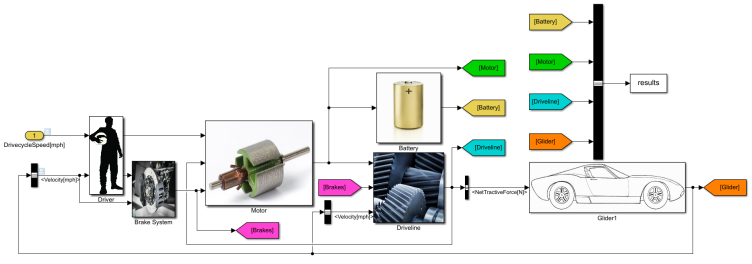
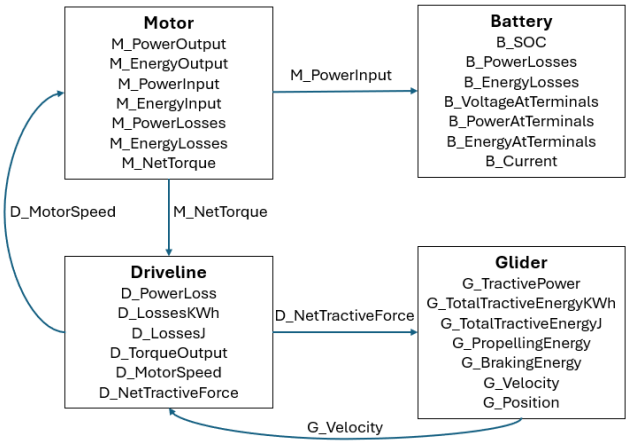


图2：电动汽车动力总成的 Simulink 模型。

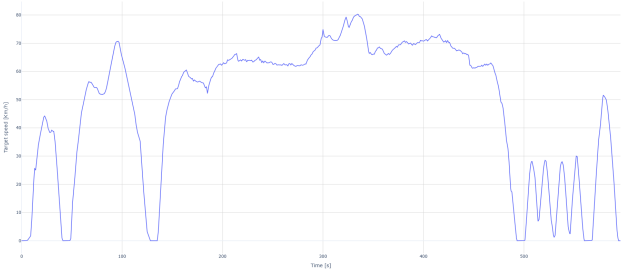
3.2. 模块版本变更的表示

在本研究中，通过有控制地改变一个具有代表性的内部参数来模拟模块版本变更。该方法能够模拟局部模型修改的影响，同时保持整体架构完全一致，这是确保仿真结果可进行一致比较的必要条件。对于每个研究模块，均根据某个关键参数对纵向动力学或车辆能耗的直接影响来选取该参数。

图 3a 展示了工作流的简化调用图，其中仅表示了版本可能发生变化的模块之间的交互作用。该图是图2 的简化版本，并突出显示了模块之间交换的流。



（a）模块调用图和变量流的简化表示。



（b）指定车辆目标速度的变化。

图3：测试用例的工作流组织和指定变量。

3.3. 关键参数的变化程度

为确定相关的扰动水平，我们开展了一项探索性研究。对选定参数施加了 5% 至 30% 的均匀变化。通过计算修改后的工作流输出与参考工作流输出之间的差值平方和，评估这些变化的影响。

本文所介绍的所有仿真均保留了两种配置。通过将其内阻提高10%来修改电池 模块，同时将其最大输出扭矩提高10%来修改电机 模块。这一选择使得测试用例同时包含一个影响易于检测的模块（电机）和一个较难识别的模块（电池）。为论证这一选择，开展了一项初步敏感性研究，在该研究中，分别对每个模块施加了不同百分比的参数修改。通过使用欧几里得距离指标，将关键变量的轨迹与参考仿真的轨迹进行比较，并结合车辆续航能力等特定领域的性能指标，对所得仿真结果进行了分析。分析表明，对电机模块的修改会产生更大且更易检测的偏差，而对电池模块的修改往往会引起系统行为中更细微的变化。其他模块，即传动系统 和滑翔机，由于未被修改，因此不应被检测出来。

3.4. 指定变量

驾驶循环构成了测试用例的主要指定变量。它定义了车辆目标速度的时间演变，并且在所有仿真中保持一致，从而能够比较不同配置的动态响应。

模块	参数	描述
电机	最大输出扭矩	电机输出的最大扭矩，影响加速能力和能量回收。 加速能力和能量回收。
电池	内阻	电池内阻，决定电气损耗 以及整体能效。
传动系统	齿轮比	决定扭矩与速度之间权衡关系的比率， 会影响机械性能。
滑翔机	滚动阻力系数	影响恒定速度下能耗的参数 速度。

表1：研究的模块及选定的代表性参数

4. 带参考的单次仿真方法（*MixED-DMD*）

所提出的方法源自 动态模态分解 (动态模态分解)[11, 12] ，并结合多个动态矩阵。其目标是选出最能解释数据的模块子集。该方法以仿真次数衡量最低成本：参考仿真加一次额外仿真，其中模块

版本根据 DoE 进行修改。该方法也可应用于多次仿真，以提高检测能力。

敏感性分析之所以称为“经典”方法，是因为它依赖成熟的工具，但对于期限紧迫的项目团队而言，其仿真成本很快就会变得难以承受。**为降低这一成本，我们采用了一个非常简单的想法：在仿真过程中更改模块版本，** X_j **和** Y_j。X_j 和 Y_j变量与控制变量类似，并成为$X_{t,\ j}$ 和$Y_{t,\ j}$。本文仅使用$X_{t,\ j}$ 变量来描述所提出的方法，但该方法同样适用于$Y_{t,\ j}$ 变量。该仿真记为 W_0（DoE），并生成数据表 T_0（DoE）。

目标是利用客户驾驶循环中大量的时间步（4000 至 18000 个），最大限度地扩大实验设计的规模。

必须定义 DoE 及其实施策略。例如：

- 对 7 个模块进行全因子设计，开展 128 次试验，可生成 128 个替换序列，每个序列持续 100 个连续时间步；
- 相同的 128 次试验设计重复两次，每次包含 50 个时间步的序列；
- 针对 10 个模块的 $2^{10} = 1024$ 次试验全因子设计，以 11 个时间步的序列实施；

序列中的时间步数必须足以揭示模块替换的影响。设计中的试验可以按任意顺序进行，甚至可以随机进行。由于动力学效应，执行顺序会改变结果$W_1 - W_0(DoE)$。如果工作流动力学复杂，或者必须检测模块之间的高阶交互作用，则理想情况下应采用全因子设计。

在所示示例中，仿真的总时长为 596 秒，时间步长为 0.1 秒；因此，生成的数据表包含 5960 个时间步。仿真通过根据四因子全因子设计修改模块版本来进行，对应于 16 种模块版本组合。

每项试验仿真 20 个时间步，并将实验设计连续重复 18 次，以覆盖仿真总时长（以及部分 19^{th} 重复）。

还测试了第二种实验设计：将相同的 16 种组合各实施两次，每种组合包含 186 个时间步。第二种设计得出的结果与第一种设计的结论相同，因此此处不再详述。

与敏感性分析一样，目标是解释数据，**但这次要进行详细解释，而不是将数据汇总为单个标量**。每个客户驾驶循环可能对应最多四次仿真：~~W_0, W_1~~, $W_0(DoE)$ 和 $W_1(DoE)$。每次仿真都会生成一个包含 18000 行（时间步）和 60 列（变量）的数据表 T。对于仿真 $W_0(DoE)$ 和 $W_1(DoE)$，数据表还包含X_t的列

$_{,j}$ 控制模块版本，因为它们由实验设计所规定。

工作流是一个动态模型，也就是说 $x_{t+1} = f(x_t)$。我们采用 动态模态分解 形式体系，并假设动力学为线性：

$$x_{t+1} = A.x_t \tag{1}$$

定义所有大小为 q 个模块（对于更一般的问题，也可扩展至大小为 q ）的子集，并记为 $\mathcal{M}_k$。对于每个$\mathcal{M}_k$，我们将 2^q 矩阵$A_{k,l}$ 与$\mathcal{M}_k$中 q 个模块所构成的完整线性模型的所有项对应起来。例如，对于子集 $\{Battery, Motor\}$，可得到三个矩阵，分别对应 $Battery$、 $Motor$和$Battery \times Motor$。

$$\begin{aligned} &\underset{A, A_{k,l}, \epsilon}{\text{Min}} \quad P_\epsilon . \sum_{t=1}^{n} \|\epsilon_t\|_2^2 + P_a . \|A\|_1 + \sum_{k,l} P_{k,l} . \|A_{k,l}\|_1 \\ &\text{under}: \quad x_{t+1} = A.x_t + \sum_{k,l \in \mathcal{T}_{all}} \delta_{t,k,l} \times A_{k,l}.x_t + \epsilon_t \end{aligned} \tag{2}$$

其中：

- $A_{k,l}$ 表示与模块子集 $\mathcal{M}_k$相关联的校正转移矩阵。索引 k 标识所考虑的模块组合，而 l 则对嵌入后与该组合相关联的不同项进行索引
- $\delta_{t,k,l} = \prod_{j \in \mathcal{M}_k}^{\text{g.}} {}_{\text{试验设计}_{t,\ j}}$ 是一个已知常数，表示时刻 t项 k, l 的激活状态；
- 符号 $\times$ 在 $\delta_{t,\ k,\ l} \times A_{k,\ l}.x_t$ 表示逐元素乘积；
- P_ϵ, P_a,$P_{k,\ l}$ 是正的标量常数，分别表示残差、参考矩阵 A 和矩阵 $A_{k,\ l}$ 的权重。选择这些权重时必须实现字典序优化：我们寻求与残差最小的解相对应的最小矩阵 A 和$A_{k,\ l}$。要求$P_{k,\ l} > P_a$，以尽量减少施加于 A 的校正。

- 特别是，与一对模块相关联的交互矩阵（例如 $A_{\text{电池、电机}}$）必须被解释为相对于各自校正矩阵 $A_{\text{电池}}$ 和 $A_{\text{电机}}$的高阶校正项。因此，其惩罚权重必须大于相应单个矩阵的惩罚权重（即 $P_{\text{电池、电机}} > P_{\text{电池}}$ 和 $P_{\text{电池、电机}} > P_{\text{电机}}$），以便优先采用基于单模块效应的简单解释，再引入交互项。

在此公式中，矩阵 $A_{k、l}$ 充当矩阵 A 的最小校正，对应于模块版本更新后的时间步。计算按组合逐一进行；在第 3 节所述的测试用例中，每对模块都需要进行一次优化。更全局的方法是在一次优化中找出最佳组合，但需要更复杂的 *PL-MIP* 公式 [13, 14]，实际效率却较低。在第 4.3 节所述的测试用例中，找出最佳模块对需要 12 秒（不包括自编码器步骤）；该步骤会将参考数据非线性嵌入潜在空间。自编码器在参考仿真数据上训练，由编码器和解码器组成：编码器将高维系统状态映射到较低维的潜在表示，解码器则从潜在空间重建原始变量。随后，该潜在表示被用作 *MixED-DMD* 流程的输入，使该方法能够捕捉数据中的非线性结构，同时在降阶空间中保持线性动力学模型。我们的公式通过一个简单且易于并行化的循环来求解全局问题。我们的 Python 实现使用 **cvxpy**包 [15] 进行线性/二次规划（该问题可以使用 L_1 或 L_2 范数求解）。

备注 1：数据表由参考表 T_0（其中所有模块控制变量均设为 0）以及根据实验设计 $T_0(DoE)$得到的表组成。这使得参考动力学能够被赋予很高的权重，并强化模块动力学的“纠正”作用。

备注 2：只要能够在测试期间修改模块版本，*MixED-DMD*方法也可应用于物理实验。例如，控制模块、映射或软件传感器都可以做到这一点。然而，显然无法在物理测试期间更换电机。L_1 版本是“稳健”版本，尤其适用于试验，因为它对测量噪声的抵抗力更强。L_2 范数的灵敏度更高，更适合检测故障程度极轻的模块，但这也意味着误检率更高。

4.1. 使用 *DMDc* 进行参考模型辨识

在应用 *MixED-DMD* 之前，我们先评估经典 动态模态分解 的检测能力。这种检测能力取决于建模质量。我们通过仅将经典 动态模态分解 应用于参考数据表，评估其适用性。实际中，必须使用*DMDc* （由 Proctor等人 [16]提出），因为指定变量（速度设定值）是控制变量。动态模型变为：

$$x_{t+1} = A.x_t + Bu_t \tag{3}$$

其中，$u_t \in \mathbb{R}^{m+1}$ 是表示速度设定值和模块激活状态（布尔变量）的控制向量，而 $B \in \mathbb{R}^{(m+1)\times p}$ 是描述它们对状态影响的矩阵。

采用*DMDc* 方法的 L_2 范数只能应用于移除冗余变量后的参考数据；最终，初始的 28 个变量中仅保留了 7 个。这 28 个变量对应模块之间交换的主要信号，并在图3a中突出显示。该图展示了模块和变量流的简化调用图。使用剩余 7 个变量获得的结果详见图2 和4。动力学表现得几乎是线性的，但事实证明，这不足以实现良好的检测性能。

谱半径 $\rho(A) = 1.0101$ 略大于 1，表明所识别的线性模型处于边缘不稳定状态。尽管其偏离 1 的幅度很小，但在长时间仿真过程中可能导致误差逐渐增大，并在一定程度上解释经典动态模态分解近似方法的局限性。

谱半径 $\rho(A) = \max|\lambda_i| = 1.0101$**矩阵 A（内部动力学）**

	B_SOC	B_PowerLosses	B_VoltageAtTerminals	M_PowerLosses	M_NetTorque	D_MotorSpeed	G_BrakingEnergy
B 荷电状态	1.000	0.000	0.001	-0.000	0.000	0.000	0.000
B_功率损耗	0.001	0.965	-0.008	0.004	-0.007	-0.566	0.000
B_端电压	-0.003	0.011	0.986	-0.014	0.004	0.235	-0.000
M_功率损耗	-0.001	-0.003	0.024	0.974	0.030	-0.229	0.003
M_净扭矩	0.002	-0.012	-0.004	0.001	0.991	-0.134	0.002
D_电机转速	0.000	0.000	0.001	-0.003	0.011	0.995	-0.000
G_制动能量	-0.001	0.001	0.002	-0.000	0.002	0.003	0.995

矩阵 B（控制）

	速度设定值
B 荷电状态	-0.000
B_功率损耗	0.564
B_端电压	-0.236
M_功率损耗	0.225
M_净扭矩	0.125
D_电机转速	0.005
G_制动能量	-0.002

表2：参考工作流的 *DMDc* 模型。

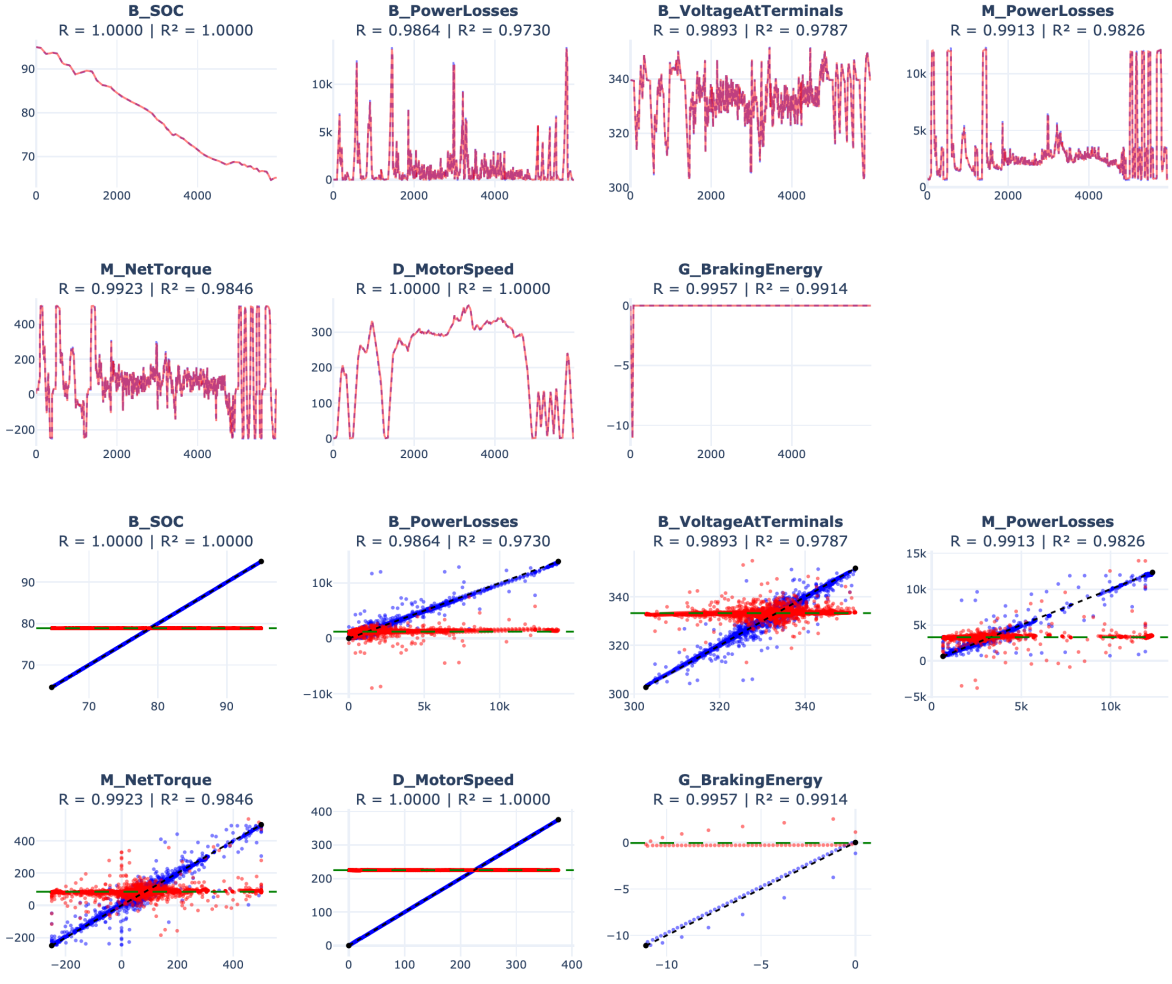


图4： *DMDc* 应用于参考数据表。时间序列图中，模型看起来几乎完美。然而，在标准质量图中可以看出，7个模型中有5个并不完美。蓝色曲线表示预测响应与真实响应的关系。红色曲线表示残差加上响应的均值。底层动力学几乎是线性的，但并非完全如此。这证明了使用嵌入是合理的（见第 4.3节），也可能解释了检测问题。

4.2. 不使用潜在嵌入的 *MixED-DMD* 模块检测

第一个应用不使用嵌入，而是使用 L_1 范数，对贪心选择相关性最低的变量后剩余的 14 个变量进行处理。详细结果见图4和5。采用 动态模态分解 方法和 L_1 范数能够较好地处理相关变量，这要归功于 PL 目标函数，该函数通过最小化矩阵 A 和 $A_{k\text{的元素绝对值}l}$；与其他变量高度相关的变量，其系数往往会被压至零。该方法正确检测出影响最大的版本变更，即电机，但在第二个变更上略有失误（差距极小；见图3），选择了传动系统 而非电池。由于传动系统 的版本并未改变，因此它不应产生任何影响。检测出传动系统 很可能是参考动态模态分解 模型质量不足所致；如图6所示，添加任意变量即可改进该模型。

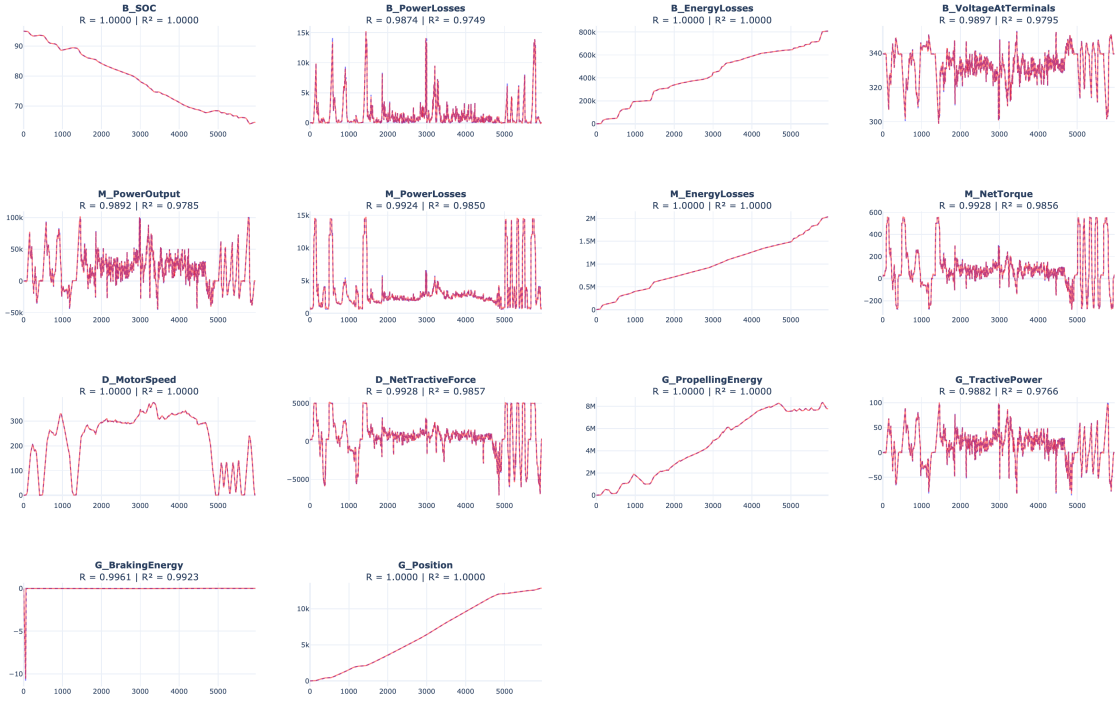


图 5：在不进行嵌入的情况下选择两个模块的最佳组合。模型几乎完美；测量曲线与预测曲线几乎处处重合。

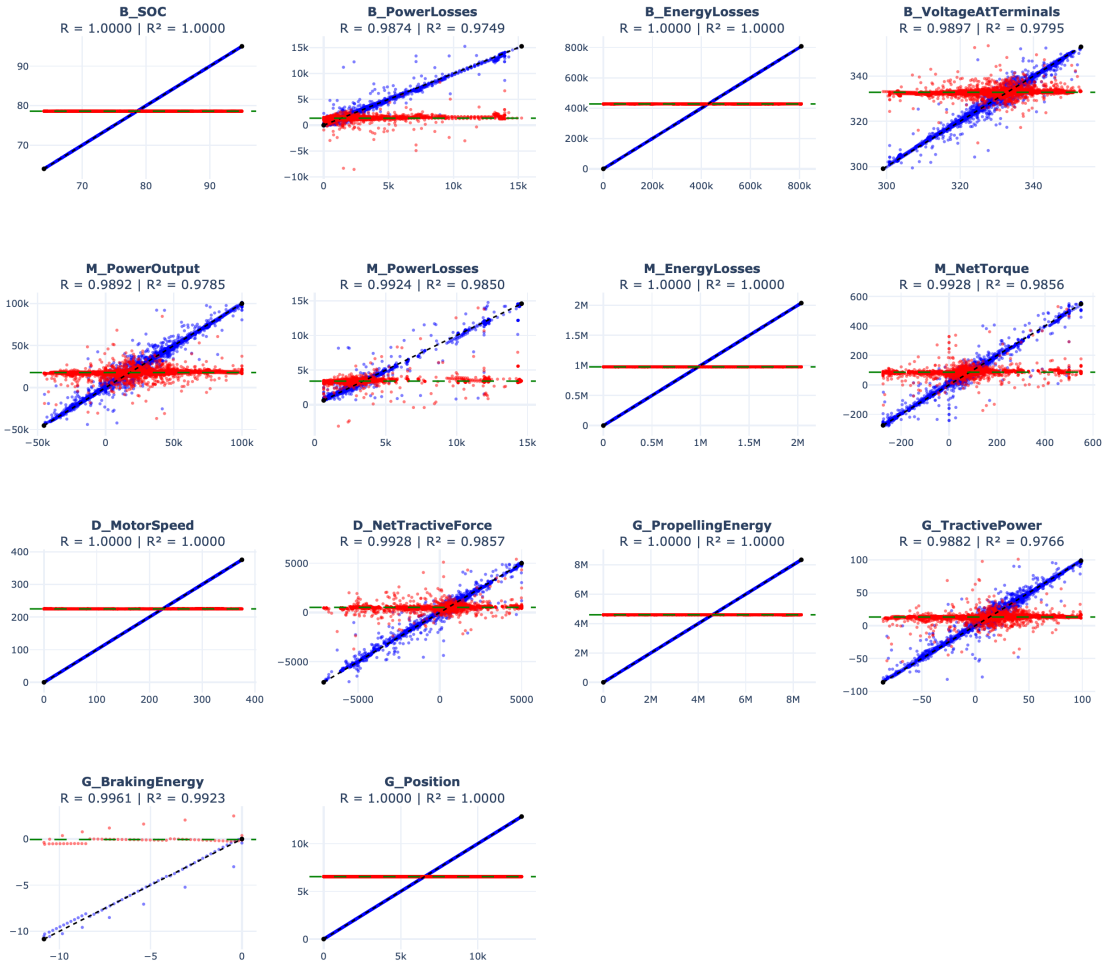


图6：未使用嵌入时两个模块的最佳组合选择。模型在基于时间的预测中表现近乎完美（见图 5），但实际上在所有变量上的表现都远非完美。与参考动态模态分解 模型相比，整体改进微乎其微。

	组合	状态	RSS	L1 Aref	L1 Am	总分	CPU
3	电机 · 传动系统	OK	846.720	8.652	18.740	8467.478	1.98秒
0	电池 · 电机	OK	847.035	12.210	25.972	8470.730	1.99秒
2	电池 · 滑翔机	OK	850.546	11.276	34.000	8505.908	1.90 秒
1	电池 . 传动系统	OK	853.002	6.303	19.544	8530.280	1.98 秒
4	电机 . 滑翔机	OK	853.384	14.653	36.040	8534.344	1.91 秒
5	传动系统 . 滑翔机	OK	855.590	8.103	23.224	8556.210	1.91秒

表3：不使用嵌入时两个模块的最佳组合选择。得分是优化目标的值，它综合了残差和矩阵元素的绝对值。

MixED-DMD 识别：电机 . 传动系统 (11.67s)

参考 (A)

B 荷电状态	1.003	-2.031	-	1.846	-1.043	-3.083	-	-1.234	0.012	-2.030	-	-1.524	-0.330	-
B 功率损耗	-	1.042	-	-0.046	0.032	0.024	-	0.013	-	0.021	-	0.026	-	-
B 能量损耗	-	-0.334	1.000	0.225	-0.070	-0.437	-	-0.135	-	-0.276	-	-0.132	-0.063	-
B 端电压	-	0.102	-	0.804	0.032	-0.026	-	-	-	0.032	-	0.045	-0.011	-
M 功率输出	-	-0.086	-	-	0.856	-0.085	-	-0.086	-	-0.107	-	-0.075	-0.010	-
M 功率损耗	-	-	-	-0.034	0.016	0.920	-	0.010	-	-0.010	-	0.017	-	-
M 能量损耗	-	-0.474	-	0.479	-0.311	-0.680	0.999	-0.224	-	-0.385	-	-0.412	-0.059	-
M 净扭矩	-	0.028	-	-0.032	0.027	0.044	-	0.947	-	0.095	-	0.045	0.007	-
D 电机转速	-	-0.008	-	-0.132	-0.119	-0.679	-	-0.298	0.998	-0.293	-	-0.098	-0.019	-
D 净牵引力	-	-0.027	-	0.031	-0.036	0.006	-	0.020	0.008	0.894	-	-0.053	-0.009	-
G 推进能量	-	-1.113	-	0.946	-0.569	-1.182	-	-0.236	0.006	-0.563	0.998	-0.745	-0.189	-
G 牵引功率	-	0.141	-	-0.129	0.134	0.010	-	0.082	-	0.118	-	1.086	-	-
G 制动能量	-	-	-	-	-	0.022	-	0.016	-	0.011	-	-0.005	1.078	-
G 位置	-	-0.136	-	0.219	-0.108	-0.820	-	-0.655	-	-0.830	-	-0.252	-0.021	1.000
可变强制目标速度	-	0.053	-	0.105	0.141	0.634	-	0.236	-	0.241	-	0.110	0.030	-

电机

B 荷电状态	-	2.817	-	-0.846	-0.213	2.575	-	0.159	-	0.304	-	-0.366	0.969	-
B 功率损耗	-	-0.235	-	0.122	-0.092	-0.081	-	-0.036	-	-0.015	-	-0.077	-0.005	-
B 能量损耗	-	0.373	-	0.018	-0.217	0.338	-	-0.030	-	-0.011	-	-0.267	0.162	-
B 端电压	-	-0.084	-	0.087	0.090	-0.024	-	0.061	-	0.075	-	0.087	-0.012	-
M 功率输出	-	0.446	-	-0.259	0.387	0.147	-	0.186	-	0.141	-	0.343	0.015	-
M 功率损耗	-	-	-	0.017	-	0.074	-	-	-	-0.019	-	-0.009	-0.011	-
M 能量损耗	-	0.767	-	-0.357	0.136	0.563	-	-0.005	-	0.066	-	0.131	0.196	-
M 净扭矩	-	-0.062	-	0.043	-0.031	-0.069	-	0.031	-	0.053	-	-0.021	-0.023	-
D 电机转速	-	-0.848	-0.006	0.552	-0.229	0.513	-0.005	0.128	-	0.134	-	-0.180	-0.032	-
D 净牵引力	-	0.070	-	-0.052	0.049	0.038	-	0.008	-	-	-	0.043	0.031	-
G 推进能量	-	1.774	-	-0.737	0.248	0.936	-	-0.105	-	0.015	-	0.161	0.643	-
G 牵引功率	-	-0.409	-	0.275	-0.260	-0.105	-	-0.123	-	-0.081	-	-0.230	-0.027	-
G 制动能量	-	-	-	-	-	-0.020	-	-0.014	-	-0.009	-	0.007	-0.100	-
G 位置	-	-0.063	-	0.219	-0.380	0.775	-	0.304	-	0.243	-	-0.395	-0.030	-
可变强制目标速度	-	0.741	0.005	-0.482	0.166	-0.484	0.005	-0.101	-	-0.110	-	0.127	-0.015	-

传动系统

B 荷电状态	-0.006	-0.681	0.006	-0.864	0.708	-1.442	0.007	-0.315	0.007	1.066	0.006	1.083	0.246	0.006
B 功率损耗	-	-0.112	-	0.068	-0.051	-0.034	-	-0.018	-	-0.026	-	-0.041	-	-
B 能量损耗	-	-0.227	-	-0.065	0.032	-0.373	-	-0.097	-	0.148	-	0.097	0.048	-
B 端电压	-	-0.300	-	0.344	-0.206	0.013	-	-0.079	-	-0.118	-	-0.199	0.011	-
M 功率输出	-	-0.089	-	0.197	-0.088	0.088	-	-0.060	-	-0.060	-	-0.120	0.007	-
M 功率损耗	-	-0.060	-	0.064	-0.042	0.013	-	-0.035	-	-0.012	-	-0.041	-	-
M 能量损耗	-	-	-	-0.268	0.215	-0.175	-	-0.120	-	0.171	-	0.271	0.043	-
M 净扭矩	-	0.024	-	-0.018	0.016	0.058	-	0.086	-	-0.065	-	-	-	-
D 电机转速	-	-0.914	-0.005	0.575	-0.259	0.334	-	0.052	-	0.075	-	-0.228	0.026	-
D 净牵引力	-	-0.023	-	0.011	-	-0.050	-	-0.054	-	0.069	-	0.013	-	-
G 推进能量	-	-0.146	-	-0.414	0.296	-0.727	-	-0.450	-	0.181	-	0.439	0.144	-
G 牵引功率	-	-0.138	-	0.086	-0.081	-0.060	-	-0.013	-	-0.034	-	-0.049	-	-
G 制动能量	-	-	-	-	-	-0.021	-	-0.013	-	-0.007	-	0.006	-0.044	-
G 位置	-	-0.305	-	-0.127	0.172	-0.162	-	0.358	-	0.584	-	0.282	0.012	-
可变强制目标速度	-	0.886	0.005	-0.563	0.254	-0.307	-	-	-	-0.041	-	0.229	-0.034	-

传动系统.电机

B 荷电状态	0.005	-	-	1.390	-1.207	1.268	-	-0.267	-0.018	-1.620	-0.006	-1.833	-0.762	-
B 功率损耗	-	0.330	-	-0.187	0.159	0.084	-	0.062	-	0.034	-	0.130	-	-
B 能量损耗	-	0.229	-	0.012	0.029	0.406	-	0.039	-	-0.187	-	-0.069	-0.127	-
B 端电压	-	0.396	-	-0.336	0.185	0.070	-	0.042	-	0.009	-	0.140	0.012	-
M 功率输出	-	-0.266	-	0.080	-0.188	-0.086	-	-0.123	-	-0.022	-	-0.125	-0.009	-
M 功率损耗	-	0.071	-	-0.069	0.050	-0.034	-	0.057	-	0.062	-	0.047	0.012	-
M 能量损耗	-	-0.289	-	0.516	-0.462	0.133	-	-0.017	-	-0.378	-	-0.612	-0.156	-
M 净扭矩	-	0.024	-	-0.013	-	0.010	-	-0.106	-	-0.111	-	0.015	0.011	-
D 电机转速	-0.005	1.413	0.006	-0.973	0.654	-0.410	0.006	0.099	-	0.102	-	0.587	0.016	-
D 净牵引力	-	-0.034	-	0.027	-0.028	-	-	0.035	-	0.045	-	-0.035	-0.021	-
G 推进能量	-	-0.575	-	0.980	-0.852	0.661	-	0.088	-0.007	-0.649	-	-1.085	-0.526	-
G 牵引功率	-	0.485	-	-0.296	0.282	0.105	-	0.151	-	0.033	-	0.191	0.022	-
G 制动能量	-0.006	0.091	0.007	-0.061	0.063	0.012	0.008	0.035	-	0.017	0.005	0.055	0.030	0.006
G 位置	-	0.624	-	-0.107	0.071	0.057	-	-0.387	-	-0.428	-	-0.075	0.046	-
可变强制目标速度	0.005	-1.306	-0.006	0.900	-0.591	0.394	-0.006	-0.125	-	-0.116	-	-0.537	0.024	-

表 4：找到的最佳双模块组合的转移矩阵。

这些结果与*DMDc* 分析一致，表明即使使用多个转移矩阵而非单个矩阵来表示，工作流动力学也并非严格线性。系统的内在非线性依然存在，并限制了检测性能。

4.3. 使用*MixED-DMD* 和潜在嵌入进行模块检测

嵌入是指对原始变量应用非线性变换，以更紧凑、结构化的方式表示和仿真复杂的车辆行为 [17, 11]。我们的方法系统地使用嵌入，以减少由非线性导致的误检。**嵌入通过在参考数据表上训练的自编码器获得**。选择自编码器是因为它易于实现和训练，且不依赖于特定仿真，例如 $W_0(DoE)$。我们逐步增大潜在空间的维度，直到获得令人满意的模型。该方法正确检测出两个更新后的模块，如图5、6和7所示。这相当于假设矩阵 $A_{k,l}$ 所表示的动态校正，在自编码器的潜在空间中是线性的。这一假设与我们的实际经验一致：模块版本变更只会轻微改变动力学。

组合	λ_{Ref}	λ_{Ref+T_1}	λ_{Ref+T_2}	$\lambda_{Ref+T_1+T2+T1*T2}$	RSS	CPU
电池 . 电机	1.012	0.998	0.998	1.000	45.183	0.09秒
滑翔机 . 电机	1.010	1.000	0.997	1.001	45.495	0.08秒
传动系统 . 电机	1.003	1.008	0.996	1.000	45.529	0.08秒
电池 . 传动系统	0.998	0.999	1.007	0.999	45.635	0.08秒
传动系统 . 滑翔机	0.998	1.007	1.000	0.999	45.764	0.08秒
电池 . 滑翔机	1.000	0.997	0.999	1.000	45.808	0.08秒

表5：双模块组合汇总。并非所有观测都能保证模型稳定性，尤其是那些仅使用工作流参考版本的观测。

参考（A）

L1	1.002	-0.006	-
L2	-0.013	1.007	0.023
L3	-	-	0.992

所选组合：电池 . 电机（0.49秒）

电池

L1	-0.009	0.010	0.014
L2	0.019	-0.011	-0.038
L3	-	-	-

电机

L1	-0.015	0.018	0.016
L2	0.014	-0.010	-0.022
L3	-0.008	-	0.013

电池·电机

L1	0.015	-0.022	-0.022
L2	-0.026	0.013	0.044
L3	0.006	-	-

表6：潜在空间中所选组合的矩阵 A、$A_{Battery}$、A_{Motor} 和 $A_{Battery.Motor}$。必须将这些矩阵相加，以获得仿真过程中使用的不同 动态模态分解模型。

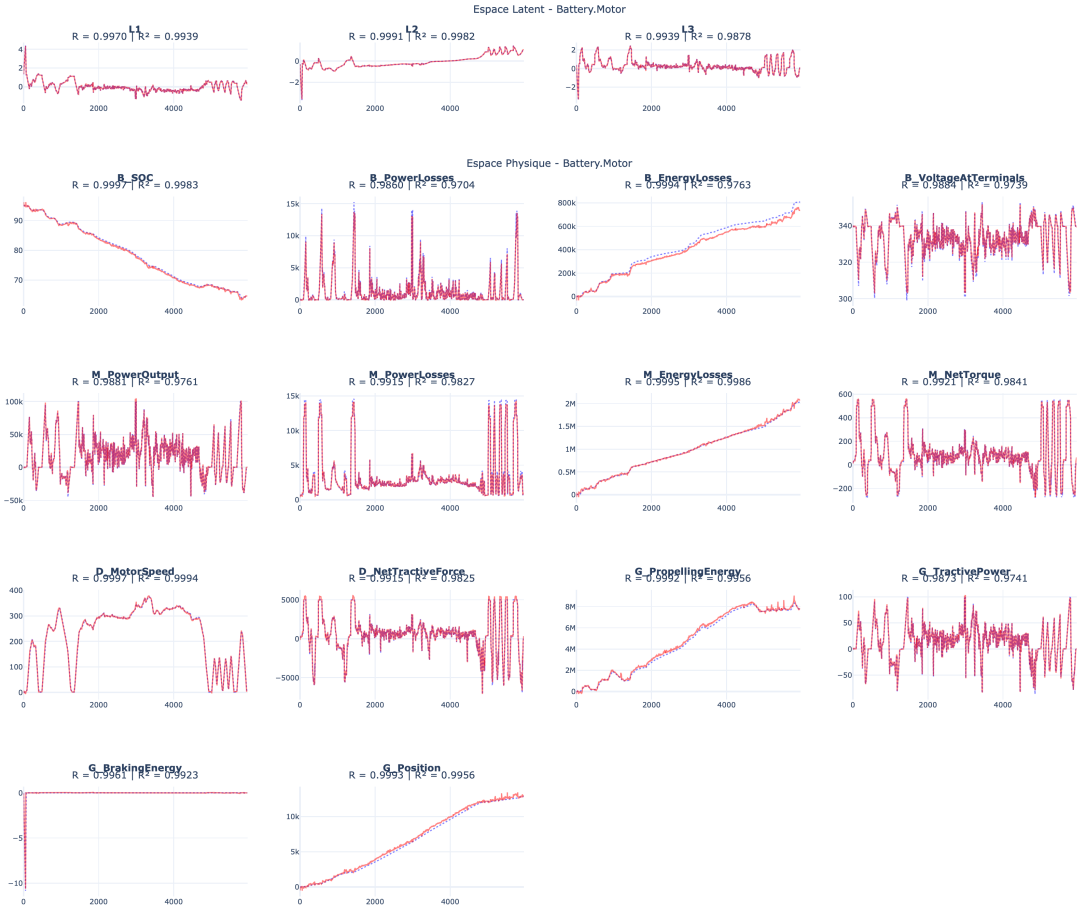


图7：参考数据表上的嵌入质量：三个维度似乎已足够（L_1 至 L_3）。模型在原始空间中看起来也令人满意。

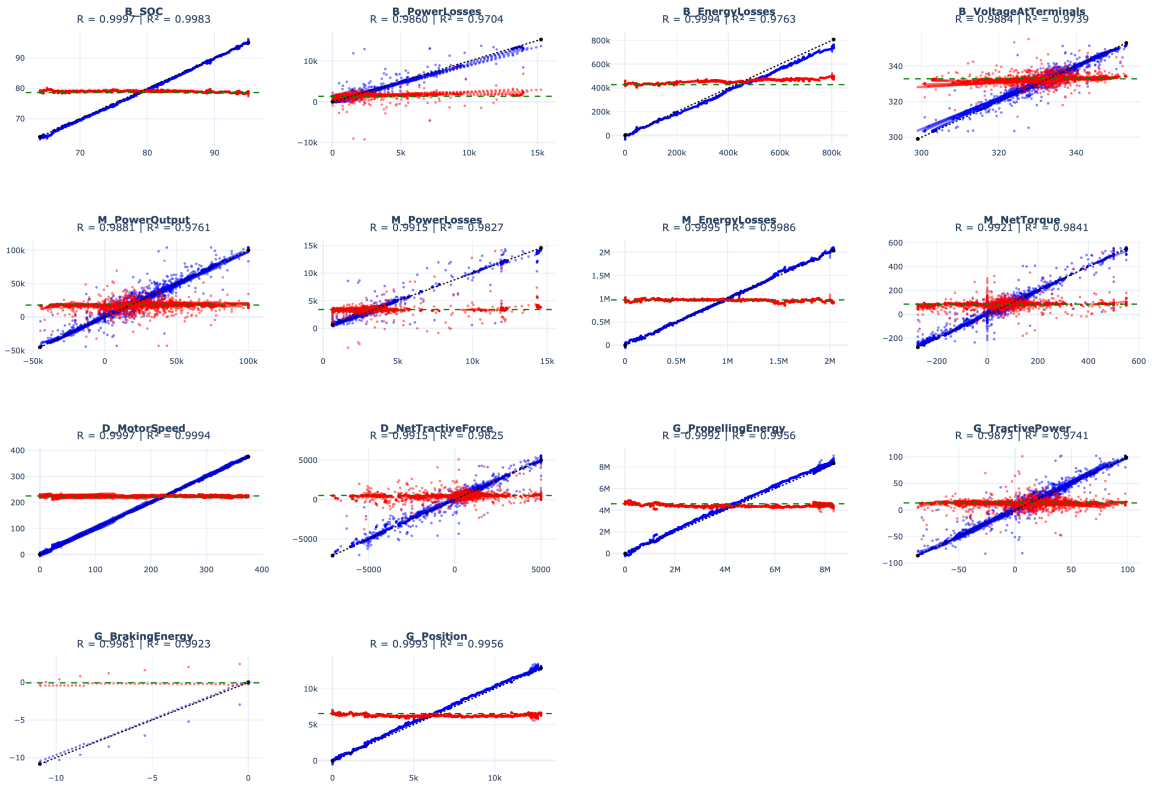


图8：可以观察到，部分观测存在较大误差，但这些误差在时间序列表示中并不明显。

5. 带参考的双仿真方法（*NoDyn*）

对动力学进行建模，以便随后检测“故障”模块，这是最直观的方法，但也是最具挑战性的方法。事实上，车辆模型动力学十分复杂，而且每次都必须找到合适的嵌入。无法保证最优嵌入总能实现线性建模。此外，工作流会随着车辆开发项目的推进而不断变化，相关研究也会反复开展。因此，需要一种尽可能自动化且“精简”的检测方法。

如果可以接受额外成本——即进行一次特定的额外仿真——就有可能几乎完全消除对动态建模的需求。这种替代方法虽然成本较高，但在对第一种方法的结果存疑时会采用。

T_0（试验设计）

t	v_1	v_2	v_3
t_1	x_{11}	x_{12}	x_{13}
t_2	x_{21}	x_{22}	x_{23}
t_3	x_{31}	x_{32}	x_{33}

逐行计算
使用 W_0 →

T_0^*

t	v_1	v_2	v_3
t_1	$\hat{x}_{11}$	$\hat{x}_{12}$	$\hat{x}_{13}$
t_2	$\hat{x}_{21}$	$\hat{x}_{22}$	$\hat{x}_{23}$
t_3	$\hat{x}_{31}$	$\hat{x}_{32}$	$\hat{x}_{33}$

图9：在 T_0(试验设计) 与 T_0^* 之间逐行计算。首先使用 $W_0(DoE)$ 进行仿真，以获得数据表 $T_0(DoE)$。然后，将 $T_0(DoE)$ 中的每一行用作初始条件，使用 W_0 进行一个时间步的仿真，从而生成 T_0^* 中对应的行。

使用线性回归进行检测：

1. 表 $T_0(DoE) - T_0^*$ 用作响应变量；
2. 指定变量（例如设定点速度）用作解释变量，因为它们在两次仿真中完全相同。它们在连续时间步之间的变化也可以纳入其中；

3. 模块的实验设计也用作解释变量（布尔变量）。对于这些布尔变量，模型必然是带交互作用的线性模型（如果设计允许）。如果实验设计是全二水平因子设计，模型可以包含所有交互作用，直至最高阶（等于模块数量）。多个模块之间显著的交互作用意味着必须同时更新多个模块，才能解释动力学的变化；

4. 模型还可以包含施加变量与布尔变量之间的交互作用。不过， DoE 的质量取决于施加的驾驶循环，以及应用布尔设计的顺序；

5. 前一时间步的响应也作为解释变量纳入其中，以表示“剩余”的动力学。实际上，实验设计的各行会重复；对于给定的模块组合，只有输入端的施加变量会发生变化，以解释输出响应。由于仅凭施加变量无法完全解释响应，因此至少还需要前一时间步的响应（也可能包括涉及这些响应的交互作用 $\mathrm{p}_{\text{响应}}$ ）必须包含。

$NoDyn$ 的检测能力受到上述第 4 点和第 5 点的限制，因为无法保证所添加的交互项足够。

以下结果对应于与$MixED\text{-}DMD$ 方法相同的测试用例：实验设计为包含 16 种组合的全因子设计，并随时间重复。实际上只有电池和电机模块得到了更新。

	传动系统	电机	电池	滑翔机
B_能量损耗	0.197	0.357	0.661	0.089
B_功率损耗	3.680	7.116	11.839	1.121
B_荷电状态	-0.000	-0.000	0.000	0.000
B_端电压	-0.005	-0.006	-0.051	-0.008
D_电机转速	0.000	-0.000	-0.000	-0.000
D_净牵引力	-0.405	-0.308	-0.486	0.345
G_制动能量	-0.001	0.000	-0.000	-0.000
G_位置	-0.000	0.000	-0.000	-0.000
G_推进能量	0.120	-0.502	-0.152	-0.196
G_牵引功率	-0.004	-0.004	0.001	0.002
M_能量损耗	0.025	1.203	-0.497	0.029
M_净扭矩	-0.013	0.277	-0.284	0.013
M_功率损耗	0.078	22.098	-9.271	-1.147
M_功率输出	7.013	26.833	-11.584	-3.419

表7：用于预测响应的模块系数。最终解释中仅保留模块系数，因为该分析仅专注于模块。

	传动系统	电机	电池	滑翔机
B_能量损耗	0.334	0.064	0.027	0.540
B_功率损耗	0.347	0.059	0.039	0.692
B_荷电状态	0.393	0.007	0.704	0.796
B_端电压	0.532	0.437	0.000	0.203
D_电机转速	0.361	0.576	0.157	0.472
D_净牵引力	0.171	0.397	0.230	0.169
G_制动能量	0.288	0.229	0.235	0.230
G_位置	0.019	0.136	0.007	0.246
G_推进能量	0.594	0.014	0.721	0.376
G_牵引功率	0.246	0.245	0.827	0.404
M_能量损耗	0.895	0.000	0.110	0.881
M_净扭矩	0.878	0.021	0.029	0.865
M_功率损耗	0.983	0.000	0.111	0.741
M_功率输出	0.490	0.041	0.459	0.687

表 8：这些系数与方差分析检验相关，可得出 p 值。 p 值越小，模块对响应的影响越大。

	传动系统	电机	电池	滑翔机
B_能量损耗	-	0.357	0.661	-
B_功率损耗	-	7.116	11.839	-
B_荷电状态	-	-	-	-
B_端电压	-	-	-0.051	-
D_电机转速	-	-	-	-
D_净牵引力	-	-	-	-
G_制动能量	-	-	-	-
G_位置	-	-	-	-
G_推进能量	-	-0.502	-	-
G_牵引功率	-	-	-	-
M_能量损耗	-	1.203	-	-
M_净扭矩	-	0.277	-0.284	-
M_功率损耗	-	22.098	-	-
M_功率输出	-	26.833	-	-

表9：分析规则非常简单：保留所有 p 值低于 0.1 的模块的非零系数。仅保留电机和电池模块，这与实际被更新的唯一模块完全一致。

在测试用例中的应用总结于图 7、8和9中。“电机”和“电池”模块显然影响最大，也恰好是仿真中实际修改过的仅有模块。

在此测试用例中，每个响应（更准确地说，每个偏差）都基于施加变量、模块布尔变量以及前一时间步的响应，使用多元线性回归进行建模（Python 函数 **ols** ，来自 **statsmodels.regression.linearmodel**），并采用选项**cov type='HC3'**。

6. 结论

我们开发了两种检测方法，*MixED-DMD* 和*NoDyn*，能够高效应对实际工业案例。现有质量工具可以检测模型更新后仿真工作流输出的差异；随后，我们的方法能够以尽可能少的仿真次数为代价，将这种差异归因于特定模块。

这些非侵入式方法成本低，可供系统性使用；随着分布式仿真的发展，它们将变得日益重要。事实上，模型创建者/供应商现在可以通过网络访问（应用程序接口）提供模型，并在不通知用户的情况下更新模型。

这些工具还可以同时应用于多个仿真，例如一次性验证多个客户驾驶循环，或利用跨模块的多种试验设计方案。

所呈现的结果表明，检测性能在很大程度上取决于基于动态模态分解的建模质量。此外，能够进行外推的精简预测工作流模型将极具价值，可用于加速仿真，例如应用于车辆中的车载应用，或通过数值优化进行设计。因此，改进参考数据表的动态模态分解建模，尤其是嵌入，成为我们的主要目标。目前的嵌入采用自编码器获得，实现起来非常简单。我们正在探索几个互补方向：

- 将所有客户驾驶循环压缩为一个驾驶循环。目标是尽可能均匀地“填充”(x_t, x_{t+1})空间。这将增大动态模态分解模型的规模，因为变量之间的相关性会降低，但应能减少外推误差；
- 在嵌入中更明确地表示控制机制，例如使用零阶样条函数表示空调的激活阈值、轨迹控制、等（见 [18, 19]）；
- 将两个步骤——嵌入与建模/检测——合并为一个阶段，如 [20], 所示，旨在迫使嵌入支持线性建模；
- 在参考嵌入的基础上，扩展各模块专属的嵌入维度。

资金支持

本研究得到 法国国家研究与技术协会的 CIFRE计划 e
（ANRT）与.....合作资助 雷诺集团。

利益冲突

作者声明，他们在本文的发表方面不存在利益冲突。

数据可用性声明

支持本研究发现的数据可根据合理请求向通讯作者索取。

作者贡献声明

Di Jiang 和 Yves Tourbier 制定了方法论，开展了仿真，并撰写了原始稿件。Sebastian Rodriguez、Herv´e Colin 和 Francisco Chinesta 为研究的概念化、结果分析以及稿件审阅和编辑作出了贡献。

参考

[1]O. C. Zienkiewicz, R. L. Taylor, 《有限元法：基础与基本原理》，（Elsevier Butterworth-Heinemann，2005），第6版 [2] N. Bordet, 《柴油燃烧的0D/1D建模：从传统模式到均质模式》，博士论文，奥尔良大学，2011，〈NNT：2011ORLE2070〉。〈tel-00717396〉[3] Y. Hammadi, 《用于优化汽车能耗的车厢热学非线性非稳态0D模型的降阶》，博士论文，巴黎文理研究大学，2020，热学 [physics.class-ph]。〈NNT：2020UPSLM027〉。〈tel-03012674v2〉[4] N. Janiaud, 《面向续航里程、性能及相关成本优化的电动汽车瞬态运行功率系统建模》，博士论文，Supélec，2011，〈NNT：2011SUPL0009〉。〈tel-00660749v2〉[5] 西门子数字化工业软件, 《雷诺借助数字孪生减少碳排放并加快汽车开发》，2023，´ 20 2025 请查阅五月 [6] T. Blochwitz, M. Otter, M. Arnold, C. Bausch, C. Clauß, H. Elmqvist, A. Junghanns, J. Mauss, M. Monteiro, T. Neidhold 等, 《用于仿真模型工具无关交换的功能模拟接口》，载于：第八届国际 2011 Modelica 会议论文集，[7] ´D. B. S. T.-G. Y. Tourbier, 《实验设计：构建与分析》，（TEC&DOC Lavoisier，1995）[8] R. Mandl, 《使用正交拉丁方测试编译器》，《Communications of the ACM》，28，1054–1058，（1985）[9] D. Kuhn, R. Kacker, Y. Lei, 《组合测试导论》，（CRC出版社，2013）[10] MathWorks, 《Simscape 电池电动汽车模型》，2022，GitHub 仓库 [11] J. N. Kutz, S. L. Brunton, B. Brunton, J. Proctor, 《动态模态分解：复杂系统的数据驱动建模》，《SIAM Review》，（2016），关于动态模态分解和 EDMD 的参考书籍与文章。书中有整章阐释了为何添加非线性观测量（乘积、平方、交互作用、样条）能够改进对 Koopman 动力学的近似。[12] P. J. Schmid, 《数值与实验数据的动态模态分解》，《Journal of Fluid Mechanics》，656，5–28，（2010）[13] G. L. Nemhauser, L. A. Wolsey, 《整数与组合优化》，（Wiley，1988）[14] D. Bertsimas, J. N. Tsitsiklis, 《线性优化导论》，（Athena Scientific，1997）[15] S. Diamond, S. Boyd, 《CVXPY：用于凸优化的嵌入 Python 的建模语言》，《Journal of Machine Learning Research》，17，1–5，（2016）[16] J. L. Proctor, S. L. Brunton, J. N. Kutz, 《带控制的动态模态分解》，《SIAM Journal on Applied Dynamical Systems》，15，142–161，（2016）[17] B. O. Koopman, 《希尔伯特空间中的哈密顿系统与变换》，《Proceedings of the National Academy of Sciences》，17，315–318，（1931），奠基性文章：提出了在观测量空间中将非线性动力学表示为线性动力学的思想，是所有现代扩展（EDMD、多项式变换等）的概念基础。[18] C. J. Stone, M. H. Hansen, C. Kooperberg, Y. K. Truong, 《扩展线性建模中的多项式样条及其张量积》，《The Annals of Statistics》，25，1371–1470，（1997），介绍了 POLYMARS。该方法基于采用自适应节点选择的多项式样条，尤其适用于高维问题和复杂交互作用。[19] J. Mounayer, S. Rodriguez, J. Tomezyk, C. Ghnatios, F. Chinesta, 《RRAEDy：非线性动力系统的自适应潜在空间线性化》，arXiv 预印本 arXiv:2512.07542，（2025）[20] J. Mounayer, S. Rodriguez, C. Ghnatios, C. Farhat, F. Chinesta, 《秩约简自编码器》，arXiv，abs/2405.13980，（2024）