研究论文上下文能改善大语言模型生成的四足机器人导航吗？一项探索性仿真研究

Barnabas
Juhasz AlpineAI
AG 达沃斯，瑞士

Rolf Pfister Lab42
AI 研究院 达沃斯，瑞士

arXiv:2611.32381v1 [cs.RO] 24 Nov 2026

摘要——**大语言模型能够编写出合格的机器人导航代码，但相关研究不仅在不断推进，模型还可能由于研究发现数量庞大而无法回忆起相关发现，并将其应用于当前任务。在本研究中，我们测试了：在大语言模型（Claude Opus 5）和运动策略保持不变的情况下，提供特定研究论文作为上下文，是否能够改善由大语言模型编写的 Go2 四足机器人的导航代码。具体而言，我们在三种设置下比较大语言模型在提供和不提供特定论文时编写的导航控制器：行为树、从共享基础编辑而来的导航器，以及从头编写的控制器。当迷宫的最短路线在回合中途被阻断时，在 Isaac Sim物理引擎下，使用一篇增量重规划论文作为上下文编写的 8 个控制器中只有 2 个完成了全部四次行程，而未提供该论文时则有 8 个中的 7 个完成（双侧费希尔精确检验，$p = 0.041$）。这一差异仅在物理仿真下出现：在理想化的运动学模拟器中，提供和不提供该论文时编写的控制器表现无明显差异。我们推测，将研究论文作为上下文提供给大语言模型后，模型往往会过度重视论文内容及其目标——在本例中即增量式重规划的效率——从而对主要任务的完成产生负面影响，并导致代码的鲁棒性降低。**

一、引言

编码智能体已经能够编写合格的机器人导航代码 [1], [2], [3], [4], [5], ，但导航仍面临尚未解决的挑战 [6], [7] ，相关研究也在持续推进，科学产出每年约增长 4% [8]。即使模型接受过这些研究的训练，也不太可能记住所有相关发现，或想到将其应用于眼前的特定任务。尤其是，模型能否可靠地回忆起某项事实，取决于该事实在其预训练数据中出现的频率 [9], [10],；而模型的实际知识截止时间往往早于报告所示的时间[11]。

因此，研究文献可以通过三种方式补充模型的知识。首先是新颖性：训练截止时间之后发表的发现，模型根本无法回忆。其次是覆盖范围：当直接提问时，模型往往会复述训练数据中最常出现的少数几种机制，而
文献中有各种各样的替代方案。第三，来源：取自论文的机制会附带其来源和可证伪的预测，而模型给出的阈值无论是否有依据，其置信度都相同。

我们关注研究论文，而非文档或代码，因为文档告诉模型如何调用接口 [12],，而论文则描述一种机制及其旨在优化的目标。尽管已知检索上下文能改善代码生成，但当上下文过长或仅部分相关时，也可能未被使用，甚至导致性能下降 [13], [14], [15]。据我们所知，研究论文上下文对在物理仿真中评估的机器人控制器所产生的净影响，尚未得到测量。

因此，我们旨在检验：在固定大语言模型和运动策略的情况下，明确向大语言模型提供研究发现，是否能改善 Go2机器人上的导航行为 [16], 。因此，我们将研究问题表述如下：

研究问题：如果向大语言模型提供来自研究文献的额外背景信息，它生成的 *Go2*机器人导航代码能否取得更高的任务完成率？

具体而言，在前两种设置中，我们以运动学模拟器中生成的 20 个迷宫世界里的目标到达率来衡量任务完成情况。在第三种也是最后一种设置中，我们评估控制器能否在 180 秒内完成全部四段单向行程，并分别在运动学模拟器和 Isaac Sim物理引擎下进行评估。

II. 方法论

在不同设置中，Go2机器人 [16]始终由同一个冻结的预训练运动策略驱动，该策略只接受速度指令。我们使用 Isaac Sim 6.0.1 随附的平坦地形运动策略（ Go2FlatTerrainPolicy），且不作任何修改。有关该运动策略和物理仿真参数，请参阅附录 D。大语言模型（ Claude Opus 5 [17]）编写了其上层的所有导航逻辑。不同设置中研究文献的具体使用和处理方法各不相同，以下小节将对此进行详细说明。

A. 作为上下文的文献

为向大语言模型提供文献背景，我们精心挑选了五篇研究论-文，以涵盖广泛的行为

特征。具体而言，我们要求每篇入选论文都描述一种可与其原始任务分离的机制，且其触发条件能够自然地出现在我们的场景中（例如盲角、位姿漂移或狭窄通道）。此外，我们特意选择了五篇广为人知、且早于模型训练截止时间（1986–2004 年）的论文。因此，虽然无法在此设置中检验新颖性，但这确保了模型对这些论文的先验知识水平较高且一致，从而可以隔离出阅读一项研究成果相较于仅仅回忆它所带来的额外作用。入选论文涵盖以下行为特征：

- 动态窗口限速 [18]
- 不可避免碰撞状态 [19]
- 蚂蚁路径积分 [20]
- 哺乳动物路径积分 [21]
- 几何重新定向 [22]

五篇论文分别被浓缩为一张卡片，作为其所描述的工作与我们其余流程之间的接口。卡片也可以被视为研究工作之上的抽象层，以及前两种设置的主要输入。具体而言，每张卡片都会描述论文的机制、其适用条件、如何将其转化为规则，以及其旨在改善的目标指标。我们采用两种方式从研究工作中生成卡片，以下将其称为不同的分组。当大语言模型仅获得论文的引文（作者、标题、出版物和年份），并根据其对该工作的记忆撰写卡片时，我们将这种设置称为分组 $\mathbb{A}$。相反，当向其提供论文全文时，我们将其称为分组 $\mathbb{B}$。在我们选取的论文中，最近的一篇发表于 2004 年，因此大语言模型很可能已在其训练数据中接触过我们的论文。这样的 A/B 对比由此使我们能够检验：阅读完整论文是否比根据模型自身对论文的记忆撰写卡片，生成更好的导航代码。我们使用 Claude Opus 5 提取卡片。有关统一卡片模式的更多细节，请参阅附录 B。最后，不同分组生成的卡片会以相同方式处理，如图 1 所示。

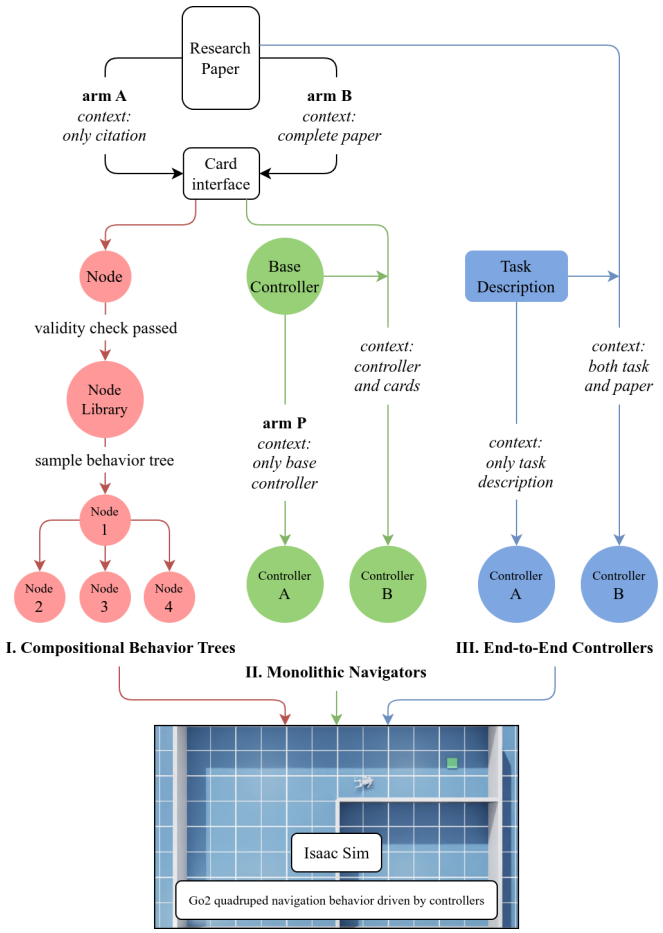


图 1。三种设置概览。在前两种设置中，每篇经过挑选的研究论文都会被压缩为一张卡片，其内容要么仅根据引文撰写（分组 $\mathbb{A}$），要么根据全文撰写（分组 $\mathbb{B}$）。**I. 组合式行为树：** 大语言模型根据每张卡片编写一个行为节点，通过有效性检查的节点会被放入一个库中，然后从中采样行为树。**II. 单体导航器：** 大语言模型使用一张卡片编辑共享基础控制器。**III. 端到端控制器：** 不使用卡片和基础代码，大语言模型仅根据任务描述，或根据任务描述与论文全文，共同编写一个完整的控制器。在所有设置中，最终生成的控制器都通过速度指令驱动 Go2机器人在 Isaac Sim [24] 中进行高层导航，其中运动策略保持冻结。

B. 组合式行为树

最初，我们的目标是让 Go2机器人的高层导航行为由 [23]由节点组成的行为树驱动。行为树非常适合我们的场景，因为它们将控制器分解为彼此独立的节点，这些节点可以独立开发和测试，然后组合成更大的行为，而无需修改彼此。具体而言，这使我们能够将每篇研究论文映射到其专属节点，从而使每篇论文的贡献保持隔离并且可归因。此外，由于行为树会在每个节拍重新评估，机器人可以通过在条件发生变化时切换行为来响应周围环境的变化。每个行为节点都封装了一种目标行为特征，只有在满足该节点所描述的条件集合时才会触发。总体而言，我们选定的 5 篇论文形成了一个包含 10 个节点的库：其中 5 个来自分组 $\mathbb{A}$，另外 5 个来自分组 $\mathbb{B}$。第一种设置经历三个阶段逐步演进，每个阶段都针对前一阶段的不足进行改进。

1）每个分组、每项任务使用独立的行为树：第一阶段中，我们通过一次调用生成行为节点。为将提取出的原理与其实现分开，我们把生成过程拆成两次调用：先提取一张卡片，再根据卡片编写节点。在这种设置下，如果节点被触发，就会完全接管机器人。随后，每个分组都会构建自己的行为树。结果表明，分组 $\mathbb{A}$ 倾向于编写完整的导航器，而分组 $\mathbb{B}$更忠实地实现论文所描述的具体机制，但往往完全无法导航。当我们从某个 $\mathbb{A}$ 导航器中移除论文所述的机制后，它在 2 次运动学展开中的成功加权路径效率（SPL）仍达到 [25]0。93 和 0。95。

2）基于共享基础节点构建行为树节点：在这一阶段，我们的目标是更好地隔离卡片及其效果。为此，我们将每张卡片都转化为一个单独的节点，并构建在共享基础导航器之上。虽然这有助于提高两个分组之间的可比性，但结果也分成了两个极端：获准使用的 10 个节点中，有 5 个从未激活，另有 5 个在 87。6%−100% 的时间步内处于激活状态。持续激活的节点实际上会覆盖基础导航器，导致下游性能变差。具体来说，我们在运动学模拟器中，针对 20 个生成的迷宫世界测量了目标到达率。基础导航器单独使用时能在 20 个世界中的 10 个到达目标，而 10 个节点无一能使其表现有所提升。

3）仅作为基础导航器扩展的节点：在第三阶段，我们将低层节点限制为只能扩展基础导航器。然而，这一限制并未带来更好的结果。10 个节点中，没有一个能在完成任务的同时改善其声称要改善的指标。例如，两个旨在减少碰撞次数的节点，只有在几乎不移动的情况下才能做到这一点，其中一个甚至从未到达目标。具体而言，在同一 20 个迷宫世界中使用运动学模拟器时，基础导航器单独运行可在 20 个世界中的 10 个到达目标；5 个分组 $\mathbb{A}$节点和 10 个分组 $\mathbb{B}$ 节点均未优于基础导航器，而其中分别有 3 个和 4 个节点使该指标变差。唯一有帮助的节点是单个分组 $\mathbb{D}$ 节点，它将目标到达率从 0.50 提高到 0.60，但代价是碰撞次数增加（从 0.40 增至 1.70）。

C. 单体导航器

与我们的行为树方法不同，在第二种设置中，我们要求大语言模型编写单体导航代码，以驱动 Go2机器人的行为。然而，这种设置需要采取额外措施，以确保不同分组之间的可比性不受影响。具体而言，我们采用共享基础导航器，这意味着各分组并非从头开始编写代码，而是修改和扩展现有的控制器。基础导航器本身由一次中立模型调用编写，既未接收卡片，也未接收论文；在运行任何分组之前，该导航器已被冻结并完成评分。在使用随机生成的 20 个迷宫世界进行测试时，它能在 45% 的环境中到达目标。其导航逻辑从未被固定，以便各分组进行改进。然而，这种设计也存在锚定效应的风险。当模型获得可运行代码时，往往倾向于进行保守修改。因此，该设置中的无效结果可能意味着卡片没有改进这一特定导航器，而不代表研究文献通常没有帮助。

随后，模型会收到提示，要求将卡片整合到基础导航器中。然而，全部 10 张卡片都将其机制描述为行为树节点。因此，在提供提示之前，我们使用替换表将卡片文本转换为行为树术语之外的表述，同时不改变任何数字，并要求模型迁移其中的原理。对于这一设置，我们还增加了一个对照臂 $\mathbb{P}$，它收到相同的请求

表 I 编辑后的导航器在 20 个迷宫世界中的目标到达率。仅基础导航器的目标到达率为 0.450，每个样本的标准误约为 0.11。

Arm	k_0	k_1	k_2	Mean	与基础导航器相比
$\mathbb{P}$	0.050	0.350	0.200	0.200	−0.250
$\mathbb{A}$	0.200	0.050	0.250	0.167	−0.283
$\mathbb{B}$	0.300	0.200	0.000	0.167	−0.283

但不提供任何卡片。将卡片整合到共享基础中，可以确保各分组的周边代码保持不变，但无法确保提示长度一致。分组 $\mathbb{B}$ 的提示长度约为分组 $\mathbb{P}$ 的两倍（$2.09\times$）。因此，$\mathbb{B}$ 与 $\mathbb{P}$ 之间的差异混合了卡片内容与单纯增加的上下文量所带来的影响。结果显示，每个分组的表现都不如单独使用基础导航器；后者在 20 个迷宫世界中的 45% 到达了目标。表 I 报告了每个分组中 $k = 3$ 个样本的目标到达率。$\mathbb{P}$ 的目标到达率为 20%，而 $\mathbb{A}$ 和 $\mathbb{B}$ 均为 16.7%；9 个控制器无一优于基础导航器。标准误约为 0.11，各分组之间的差异处于噪声范围内。

D. 端到端控制器

我们设计的第三种也是最后一种设置，旨在消解先前方法所得混合结果的歧义。在这一设置中，没有卡片，也没有供模型据以锚定的基础代码。Go2机器人 [16] 的任务是在迷宫式环境中到达目标位置，返回起点，然后重复这一过程。然而，环境会在每个回合中发生一次变化。第二次行程结束后，机器人位于起点时，最短路线上的一扇门关闭，机器人被迫寻找一条前往目标位置的新路线。环境如图 2所示；有关环境的确切设置和参数，请参阅附录 C。作为研究论文，我们选择了 D* （动态 A*） [26], 这一知名的增量式重新规划方法。它会根据截至目前所见全部信息构建的地图，引导 Go2 沿最短路径行进；每当传感器发现该路径上存在障碍物时，它就会更新地图并绕开障碍物重新规划。其主要优势在于效率，因为它会在局部修复现有计划，而不是从头开始重新规划。虽然 D* 很自然地适用于我们的任务，但它早在模型训练截止时间之前数十年就已提出，而且模型仅凭任务描述就能编写完整的控制器。因此，这篇论文并非在引入新知识，而是在检验：将模型已经知晓的研究发现置于上下文中，是否会改变模型将该发现应用于任务的方式。为处理运行间方差，我们为每个分组生成 $k = 8$ 个控制器。每个控制器都是基于相同提示词单独生成的结果。首先在运动学模拟器中评估这些分组，此时它们的表现无法区分（$5/8$ 对比 $6/8$, $p = 1.000$）。随后，我们在 Isaac Sim物理引擎下评估相同的 16 个控制器，其中 $7/8$个不含论文引用的控制器完成了全部四次行程，而含有论文引用的控制器中有 $2/8$ 个完成了全部四次行程（$p = 0.041$）。只有当机器人完成全部行程时，控制器才算成功

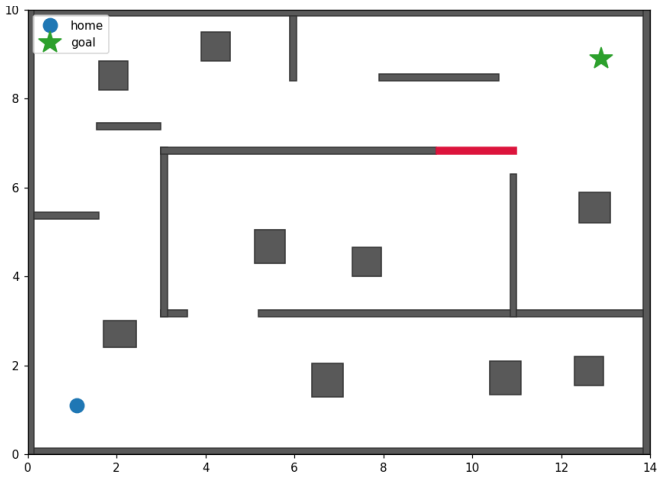


图 2. 第三种设置下的环境。Go2机器人需要找到目标位置（绿色星形），返回起点（蓝色圆形），并重复这一过程。完成第二次行程后，会出现一个障碍物（门），挡住此前最快的路线（以红色标出）。

表 II 每个分组的 8 次独立生成中完成全部 4 次行程的控制器数量。使用双侧费希尔精确检验比较各分组。

	不带论文	带纸张	费希尔 p
运动学模拟器	5/8	6/8	1.000
Isaac Sim物理引擎	7/8	2/8	0.041

四次单向行程，并且在分配的 180 秒时间预算内完成。用于生成控制器的两个提示词除了论文引用本身之外完全相同，且都没有提及环境的挑战性。两种生成设置之间的确切差异见附录 A。表 II 报告了每个分组中 8 个控制器里完成全部四次行程的数量，而图 3展示了物理仿真下每个分组的一个控制器示例。在物理仿真下，不含论文引用的控制器平均完成 3.75 次行程，而含有论文引用的控制器平均完成 2.38 次。在示例配对中，最短路径被阻断后，不含论文引用的控制器找到了绕行路线，而含有论文引用的控制器陷入停滞，始终未能完成第三次行程。然而，这并不能概括所有失败情况，因为有一个含有论文引用的控制器在第二次行程时就已失败，当时路径尚未被阻断。此外，两个模拟器对 16 个控制器中的 6 个给出了不同结论：两个不含论文引用的控制器在运动学模拟中失败、但在物理仿真下成功；而四个含有论文引用的控制器在运动学模拟中成功、但在物理仿真下失败。附录 F展示了每个分组中较不常见结果的一个示例。

III. 讨论

还需要进行更多实验和消融研究，才能确切确定为什么额外的文献上下文没有带来更好的结果。目前，我们提出了两个主要假设。

假设1：模型倾向于采用研究论文自身的目标。在第三种设置中，论文的主要贡献在于效率，因为它会在本地修复现有计划，因此随着地图不断扩大，重新规划的成本仍然较低。我们的试验环境可能还不够复杂，无法体现这一优势；与此同时，论文却在代码中引入了复杂性，并将模型的注意力引向其自身的优先事项。我们基于编辑的方法中的卡片也显示出类似趋势。每张卡片都会预测相对于指定基线的指定指标有所改善，而对每个使用该卡片的控制器进行评分后，共得到60项预测。相对于卡片自身指定的基线，60项中有43项（72%）得以实现。然而，相对于控制器实际替代的基础导航器，只有10项预测得以实现，而且在这些情况下，控制器达到目标的频率反而更低。换言之，没有一张卡片能够在不损害任务表现的情况下改善其指标。

假设2：受论文影响编写的代码在模拟物理环境下更加脆弱。在参考论文编写的八个控制器中，有三个仅以增量方式实现了 D*，没有回退到完全重新规划的机制；这三个控制器在物理仿真下全部失败，尽管另外两个失败的控制器确实包含完全重新规划。一种可能性是，经过局部修复的搜索树可能会变得过时，使机器人继续沿着一条已不存在的路线行进；而每一步都重新计算所有内容的代码则无法保留过时的信念。另一种可能性是，不参考论文编写的控制器通过模拟短时滚动模拟来选择速度。当步态未能完全按照指令行进时，这种方法具有更好的适应性；而沿着 D* 的回溯指针行进则没有这种效果，因为这些回溯指针将每个地图单元格连接到通往目标的路径上的下一个单元格。

IV. 结论

总而言之，在我们测试的设置中，加入研究论文上下文并未提升导航性能，因为由卡片构建的 10 个节点和所有编辑后的导航器均未超过共享基础导航器。在我们的第三种设置中，各分组在运动学模拟器中的表现无法区分。然而，在物理仿真下，参考论文编写的控制器完成全部四次行程的频率低于未参考论文编写的控制器（2/8 对 7/8、$p = 0.041$）。因此，我们对研究问题的回答如下：

在我们测试的任务、论文和模型范围内，提供研究论文上下文并未提高由大语言模型编写的导航代码在模拟 *Go2*机器人上的任务完成率，在某些情况下反而降低了任务完成率。

V. 局限性与效度威胁

局限性。 评估流程的设计受可比性所限，这是不可避免的。尤其是在整个研究过程中，如何在保持与基线可比性的同时，使研究范围仍与未来的现实世界应用相关，始终是一项挑战。在当前的智能体开发环境中，模型会处理一系列长期且不断变化的

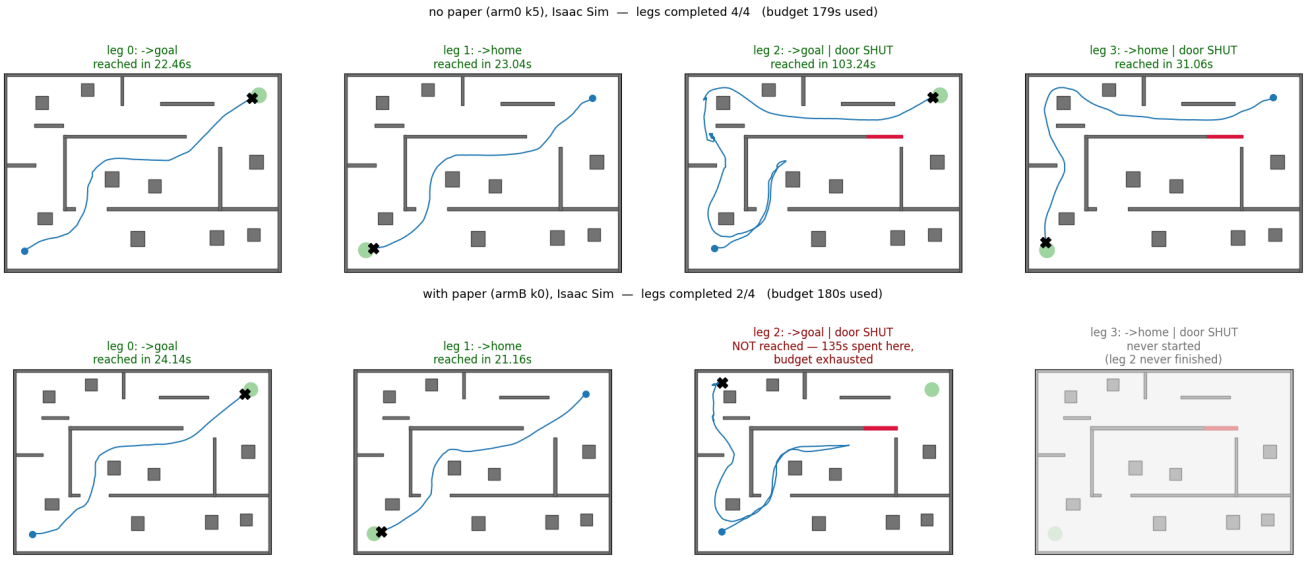


图 3. Isaac Sim 中两个控制器的路径，每个行程对应一个面板。上：不提供论文（4/4 个行程）。下：提供论文（完成 2/4 个行程，第三个行程时预算耗尽）。第三个行程中阻挡最短路线的障碍物以红色突出显示。

指令，通常还会与其他智能体协作，或与人类保持互动。然而，由于不同运行之间的输出方差很大，这种设置将不再与基线具有可比性。

有效性威胁。 由于无法设置温度参数，模型输出具有非确定性。在保持模型配置和提示词不变的情况下，我们观察到生成的控制器长度的输出方差为 62%（354 行）。我们应对方差的主要方法是汇总每个分组中多个生成的控制器的结果，从而降低高方差对统计分析的影响。此外，从设计角度来看，我们的若干设置都以共享基础为基础构建控制器。这有助于提高可比性，因为只有在新实现确有需要时，智能体才会修改现有的共享代码。在行为树的情况下，智能体甚至无法修改这个基础导航器，因为其他所有节点都是直接构建在其之上的。

VI. 未来工作

我们在第三节讨论的假设指向了三个未来研究方向。首先，未来工作可以探索在更复杂的环境中开展实验（例如，在更大的房间中设置更多障碍物，并采用长时域目标），在这种情况下，研究论文带来的效率提升或许会更有意义。此外，加入回退到完全重新规划的机制，将直接检验分组 $\mathbb{B}$ 的脆弱行为是否源自局部修复的规划。最后，使用在当前模型发布后发表的研究论文来构建库，可能有助于揭示该流程的潜在优势。

致谢

本研究得到 Mindfire 基金会的支持。我们感谢他们提供资金，并感谢他们在整个项目期间给予的反馈。

生成式人工智能声明

作者感谢 Anthropic Claude 和自主开发智能体在仿真及评估脚本的代码框架搭建、自动化文献交叉核对以及稿件语法润色方面提供的协助。作者审阅并核实了所有人工智能辅助贡献和参考文献条目的准确性。所有分析、实验设计和最终结论均由作者撰写并核实。

参考文献

[1] J. Liang、W. Huang、F. Xia、P. Xu、K. Hausman、B. Ichter、P. Florence 和 A. Zeng，“代码即策略：用于具身控制的语言模型程序”，载于 *2023 年 IEEE 国际机器人与自动化会议（ICRA）*，2023 年，第 9493–9500 页。[2] C. Huang、O. Mees、A. Zeng 和 W. Burgard，“用于机器人导航的视觉语言地图”，*2023 年 IEEE 国际机器人与自动化会议（ICRA）*，第 10 608–10 615 页，2022 年。 [在线]。可访问：https://api.semanticscholar.org/CorpusID:252846548

[3] S. Vemprala、R. Bonatti、A. Bucker 和 A. Kapoor，“用于机器人学的 ChatGPT：设计原则与模型能力”，2023 年。 [在线]。可访问：https://arxiv.org/abs/2306.17582。L. Fu、J. Yu、K. El-Refai、E. Kou、H. Xue、H. Huang、W. Xiao、G. Wang、D. Niu、F.-F. Li、G. Shi、J. Wu、S. Sastry、Y. Zhu、K. Goldberg 和 L. J. Fan，“Cap-X：用于基准测试和改进机器人操作编码智能体的框架”，2026 年。 [在线]。可访问：https://arxiv.org/abs/2603.22435。K. Elmaaroufi、J. Svegliato、S. Kalade、G. Schelle、S. A. Seshia 和 M. Zaharia，“Rho：你的编码智能体其实是一名机器人学家”，2026 年。

[4] [5] [6] 2606.16458 在线。可访问：https://arxiv.org/abs/。C. Cadena、L. Carlone、H. Carrillo、Y. Latif、D. Scaramuzza、J. Neira、I. Reid 和 J. J. Leonard，“同时定位与建图的过去、现在与未来：迈向鲁棒感知时代”，*IEEE《机器人学汇刊》*，第 32 卷，第 6 期，第 1309–1332 页，2016 年 12 月。[在线]。可访问：http://dx.doi.org/10.1109/TRO.2016.2624754

[7] X. Xiao、B. Liu、G. Warnell 和 P. Stone，“使用机器学习进行移动机器人导航的运动规划与控制：综述”，2022 [ ] 2011 13112 。在线。可访问：https://arxiv.org/abs/。[8] L. Bornmann、R. Haunschild 和 R. Mutz，“现代科学的增长率：一种潜在分段增长曲线方法，用于对来自既有和新建文献数据库的出版物数量进行建模”，Humanities and Social Sciences Communications，第 8 卷，第 224 页，2021 年 10 月。[9] N. Kandpal、H. Deng、A. Roberts、E. Wallace 和 C. Raffel，“大语言模型难以学习长尾知识”，载于《第 40 届国际机器学习会议论文集》，ICML'23 系列。JMLR.org，2023 年。[10] A. Mallen、A. Asai、V. Zhong、R. Das、D. Khashabi 和 H. Hajishirzi，“何时不应信任语言模型：参数化记忆与非参数化记忆有效性的研究”，载于《计算语言学协会第 61 届年会论文集》（第 1 卷：长篇论文），A. Rogers、J. Boyd-Graber 和 N. Okazaki 编。加拿大多伦多：计算语言学协会，2023 年 7 月，第 9802–9822 页。 [在线]。可访问：https://aclanthology.org/2023.acl-long.546/ [11] J. Cheng、M. Marone、O. Weller、D. Lawrie、D. Khashabi 和 B. V. Durme，“过时数据：追踪大语言模型的知识截止时间”，2024 [ ] 2403 12958 。在线。可访问：https://arxiv.org/abs/。[12] S. Zhou、U. Alon、F. F. Xu、Z.-R. Wang、Z. Jiang 和 G. Neubig，“文档提示：通过检索文档生成代码”，载于《国际机器学习表征会议》，2022 年。 [在线]。可访问：https://api.semanticscholar.org/CorpusID:252734952 [13] Z. Z. Wang、A. Asai、X. V. Yu、F. F. Xu、Y. Xie、G. Neubig 和 D. Fried，“CodeRAG-bench：检索能否增强代码生成？”，载于《计算语言学协会会议论文集：NAACL 2025 研究成果》，L. Chiruzzo、A. Ritter 和 L. Wang 编。美国新墨西哥州阿尔伯克基：计算语言学协会，2025 年 4 月，第 3199–3214 页。 [在线]。可访问：https://aclanthology.org/2025.findings- naacl.176/ [14] F. Shi、X. Chen、K. Misra、N. Scales、D. Dohan、E. Chi、N. Sch¨arli 和 D. Zhou，“大语言模型很容易受到无关上下文的干扰”，载于《第 40 届国际机器学习会议论文集》，ICML'23 系列。JMLR.org，2023 年。[15] Y.-F. Du、M.-Y. Tian、S. Ronanki、S. Rongali、S. Bodapati、A. Galstyan、A. Wells、R. Schwartz、E. A. Huerta 和 H. Peng，“即使检索完美，仅上下文长度也会损害大语言模型性能”，ArXiv，第 abs/2510.05381 卷，2025 年。 [在线]。可访问：https://api.semanticscholar.org/CorpusID:281820429 [16] 2 宇树科技，“Go 四足机器人”，https://www.unitree.com/go2。[17] Anthropic，“Claude Opus 5”，https://www.anthropic.com/claude，2026 年。[18] D. Fox、W. Burgard 和 S. Thrun，“用于碰撞避免的动态窗口法”，IEEE Robotics Autom. Mag.，第 4 卷，第 23–33 页，1997 年。 [在线]。可访问：https://api.semanticscholar.org/CorpusID:1718553 [19] T. Fraichard 和 H. Asama，“不可避免碰撞状态——迈向更安全的机器人？”Advanced Robotics，第 18 卷，第 10 期，第 1001–1024 页，2004 年。[20] M. M¨uller 和 R. Wehner，“沙漠蚂蚁 Cataglyphis fortis 的路径积分”，Proceedings of the National Academy of Sciences，第 85 卷，第 14 期，第 5287–5290 页，1988 年。 [在线]。可访问：https://www.pnas.org/doi/abs/10 1073 85 14 5287 。/pnas。。。A. S. [21] Etienne 和 K. J. Jeffery，“哺乳动物路径积分”，Hip- pocampus，第 14 卷，第 2 期，第 180–192 页，2004 年。[22] K. Cheng，“大鼠空间表征中的纯几何模块”，Cognition，第 23 卷，第 2 期，第 149–178 页，1986 年。 [在线]。可访问：https://www.sciencedirect.com/science/article/pii/0010027786900417 [23] M. Colledanchise 和 P. Ogren，《机器人学与人工智能中的行为树：导论》。CRC出版社，2018 年。[24] NVIDIA，“Isaac Sim，6.0.1 版”，https://developer.nvidia.com/isaac/sim，2026 年。[25] P. Anderson、A. Chang、D. S. Chaplot、A. Dosovitskiy、S. Gupta、V. Koltun、J. Kosecka、J. Malik、R. Mottaghi、M. Savva 等，“具身导航智能体的评估”，arXiv 预印本 arXiv:1807 06757，2018 年。。[26] A. Stentz，“部分已知环境中的最优且高效的路径规划”，载于《1994 年 IEEE 国际机器人与自动化会议论文集》。IEEE，1994 年，第 3310–3317 页。

附录 A 提示词模板与上下文注入

第三种设置中的每个控制器，都是通过 Claude Code CLI（版本 2.1.278）调用 Claude Opus 5（模型 ID 为 claude-opus-5）一次后生成的。本附录展示了第三种设置的两个分组之间的差异，也就是研究论文被注入提示词的区块。两个分组接收相同的任务描述；此外，有纸分组还会收到下方区块，其中 {{TASK}} 标记任务描述，{{PAPER}} 标记以纯文本呈现的论文。移除整个区块即可得到无纸提示词，因此除此之外，这两个提示词逐字节完全相同。

{{TASK}}

<!-- ---- 论文区块开始 ---- -->

来源材料

下方重现了一篇已发表论文的全文。**请运用论文确立的内容。**阅读论文，梳理其中哪些观点与上述任务相关，并在有帮助的地方将其融入你的控制器。

依据文档开展工作，而不是依赖你对文档的记忆。如果两者有出入，以文档为准。这篇论文讨论的并非配备平面激光雷达的 Go2，因此如何将其内容迁移应用是你的任务——保留可迁移的部分，不要强行套用不适用的内容。

在文件顶部添加一条简短注释，说明你从论文中借鉴了什么，以及相关内容在论文中的出处。

“‘
{{PAPER}} “‘

<!-- ---- 论文区块结束 ---- -->

附录 B 卡片模板与示例卡片

每个分组使用相同的卡片模式，该模式包含六个字段，如表 III 所示。此外，每张卡片还包含一个标识符 和一个节点名称。将卡片转换为代码的提示词不会获知卡片来自哪个分组。

表 III 统一原则卡片模式。

字段	内容
来源	引文，以及文本跨度：阐述该机制的章节、页面、图或方程；若作者无法指出具体位置，则留空。模型撰写的文本跨度属于一项主张，并标记为未经验证。
机制	发生了什么，以及它为何有帮助：研究发现，而非实现。
触发条件	适用条件，即成为前置条件的条件。
推导	研究发现如何转化为适用于这台机器人和这些传感器的规则，包括数值、单位以及其背后的推理。
预测	陈述（可证伪的主张）、指标（评分器从封闭列表中计算得出）、基线、最小效应 _（足以构成支持的最小效应）和成本指标（预期会变差的指标）。
置信度	高／中／低，具体定义以来源中的说明为准

附录 C 第三种设置环境配置

表 IV 列出了任务、世界和控制器接口。所有长度均使用基准测试自身的几何代码计算，计算时将障碍物按 0.30 米的机器人半径进行膨胀。两个仿真器使用相同的二维世界描述。

表 IV 第三种设置的任务、世界与控制器接口。

设置	数值
世界	手工绘制的建筑， 14×10米，墙壁厚 0.15 米；未向控制器
起点，目标	$(1.10, 1.10)$ 和 $(12.90, 8.90)$m
Task	四段单向行程：起点 $\rightarrow$ 目标，目标 $\rightarrow$ 起点，起点 $\rightarrow$目标，目标 $\rightarrow$ 起点。该测试框架设置当前目标，并在抵达时切换目标。重置()只调用一次，在第 1 段行程之前
门事件	最短路线上的一扇门（1.80×0.15米）在行程1–2期间保持开启，并在行程2结束后、机器人位于起点时关闭一次
最短路线	门开启时为15.03米；门关闭时为19.36米（+4.33米）。绕行路线最窄处的可用宽度为0.77米，机器人中心可在其中通过
时间预算	四次行程合计 180 秒的模拟时间
到达半径	当前目标周围 0.40 米
成功	四次行程均在时间预算内完成
控制频率	50 Hz（每 0.02 秒调用一次 step()）
激光雷达	在 360° 上使用 181 束，最大量程为 6.0 米（∞ 无返回值），通过针对两个模拟器中的二维世界进行光线投射以解析计算
Pose	精确的位姿估计（x， y， ψ），无漂移。位姿误差估计为 0
其他观测	回合时间、接触标志和当前目标点
指令	机体坐标系（v_x， v_y， ω_z），饱和限制为 $\lvert v_x \rvert$， $\lvert v_y \rvert \leq 0.8$米/秒和 $\lvert \omega_z \rvert \leq 1.0$弧度/秒
允许导入的模块	Python 标准库（不包括进程、文件系统和网络模块）、NumPy，以及指令类型。任何其他类型都会在控制器运行前被拒绝
运动学模拟器	机体坐标系速度以 50 Hz 的频率精确积分；机器人是半径为 0.30 m 的圆盘。
物理模拟器	仅当某一步会使中心点与障碍物的距离小于 0.165 m 时，该步才会被拒绝。起点位姿为起点，航向角为 0.6 rad。每个控制器重复三次在环路中使用冻结的 *Go2FlatTerrainPolicy* ，并采用 Isaac Sim 6.0.1（附录 D）；机身高度为 0.70×0.31 米。试点中每个控制器运行一个回合

附录 D

物理仿真参数

表 V 列出了我们的代码为物理仿真运行设置或记录的内容。未列出的所有内容均保留 Isaac Sim Go2 策略示例和机器人资产的默认值。

表 V 由我们的代码设置或记录的物理仿真设置。

设置项	数值
仿真器	NVIDIA Isaac Sim 6.0.1，Python 3.12
运动策略	*Go2FlatTerrainPolicy*，随 Isaac Sim 提供的预训练平坦地形策略模块 *isaacsim.robot.policy.examples.robots.go2*
机器人资产	来自 Isaac Sim 资产库的 Isaac/Robots/Unitree/Go2/go2.usd
物理步长	0.005 秒（200 Hz）
降采样	4：每个速度指令保持 4 个物理步
生成与稳定	在起点上方 0.55 米处生成，朝向为 0.6 弧度。此外，机器人会接收 2.0 秒的零指令以站立，然后回合时钟才开始计时
场景	从二维平面图导出
Door	单个图元：在第 1–2 次行程中移至地面下方，并在第 3–4 次行程中移回原位
感知	激光雷达通过射线投射与二维世界进行解析计算。位姿从仿真机器人基座读取。不使用渲染传感器
渲染	评分运行时关闭；渲染绝不会推进物理仿真
硬件	本地工作站，NVIDIA RTX 4060 笔记本电脑 GPU（8 GB）
挂钟	每个 180 秒回合耗时 170–305 秒

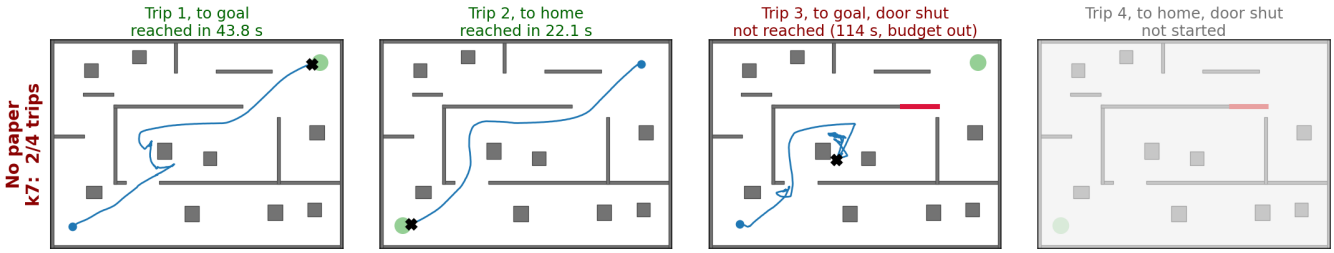
附录 E 各控制器结果

表 VI 列出了第三种设置中的每个控制器。在两种模拟器中判定结果不同的六个控制器分别是无纸 k2 和 k5（仅在物理仿真中成功），以及有纸 k1、k2、k6 和 k7（仅以运动学方式成功）。

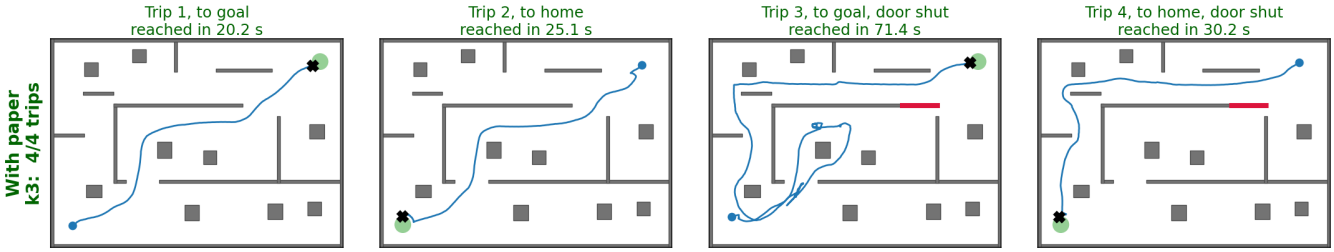
表 VI 第三种设置中各控制器完成的行程，包括运动学重复实验中的三次行程和表 II 中的单次物理实验轮次。行数：生成文件的长度。粗体：全部四次行程，计为成功。为供参考，列出了先前设置中的共享基础导航器。

Arm	k	行数	运动学（三次重复实验）	物理仿真
无论文	k0	658	**4**,**4**, **4**	**4**
	k1	770	**4**, **4**, **4**	**4**
	k2	815	3, 3, 3	**4**
	k3	651	**4**, **4**, **4**	**4**
	k4	760	**4**, **4**, **4**	**4**
	k5	740	2, 2, 2	**4**
	k6	851	**4**, **4**, **4**	**4**
	k7	758	3, 3, 3	2
	成功次数		5/8	7/8
带纸张	k0	869	2, 2, 2	2
	k1	781	**4**, **4**, **4**	2
	k2	922	**4**, **4**, **4**	2
	k3	659	**4**, **4**, **4**	**4**
	k4	568	**4**, **4**, **4**	**4**
	k5	599	2, 1, 2	2
	k6	636	**4**, **4**, **4**	1
	k7	888	**4**, **4**, **4**	2
	成功次数		6/8	2/8
共享基础	–	577	2, 2, 2	3

图 3 展示了物理仿真下每个分组最常见的结果，图 4 则补充展示了不太常见的结果。(a) 中的控制器是 8 个未参考论文编写的控制器中唯一一个在物理仿真下失败的控制器。它完成了前两次行程，但门关闭后无法完成第三次行程。在运动学模拟器中，同一个控制器完成了四次行程中的三次。(b) 中的控制器是两个在物理仿真下成功的参考论文编写的控制器之一，完成了全部四次行程；在运动学模拟器中它同样完成了全部四次行程。



(a) 未参考论文编写：完成 4 次行程中的 2 次。



(b) 参考论文编写：完成 4 次行程中的 4 次。

图 4. 每个分组不太常见结果在 Isaac Sim 物理仿真下的轨迹。