# RoboTool: Creative Robot Tool Use with Large Language Models


Mengdi Xu*,+,1, Peide Huang*,1, Wenhao Yu*,2, Shiqi Liu[1], Xilun Zhang[1], Yaru Niu[1], Tingnan Zhang[2], Fei Xia[2], Jie Tan[2], Ding Zhao[1].

* Indicates equal contribution. + Work partially done at Google DeepMind. [1] Carnegie Mellon University. [2] Google DeepMind. Contact: mengdixu@andrew.cmu.edu

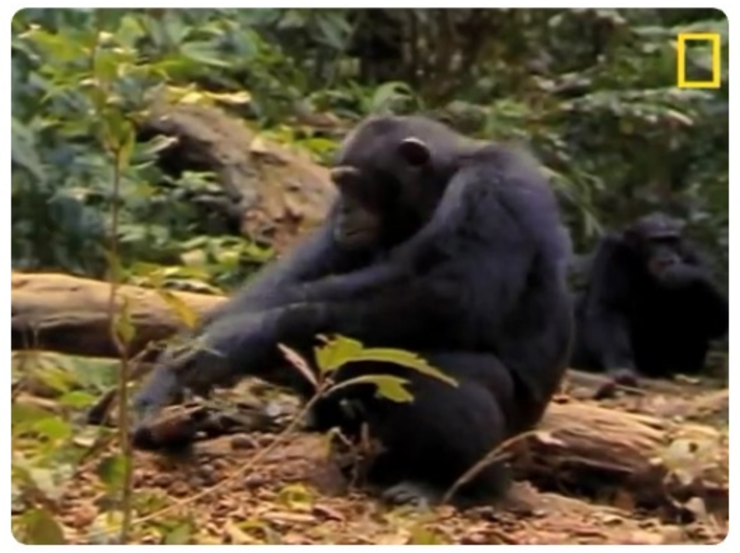

Carnegie Mellon University Google DeepMind

Scan for full paper!

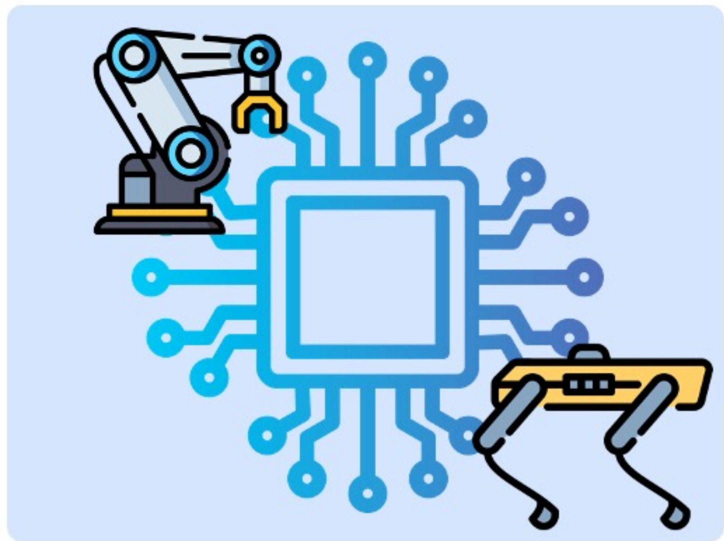
## Motivation

Tool use is a hallmark of intelligence. How to enable robots use tools creatively?

Are Crows the Ultimate Problem Solvers?
Source: BBC Earth



Apes with Tools
Source: National Geographic



Can robots finish similar tasks?

**Creative tool use** provides a solution to long-horizon task and motion planning problems with environment- and embodiment-related constraints.

## RoboTool Framework

**We propose RoboTool built on large language models (LLMs).**

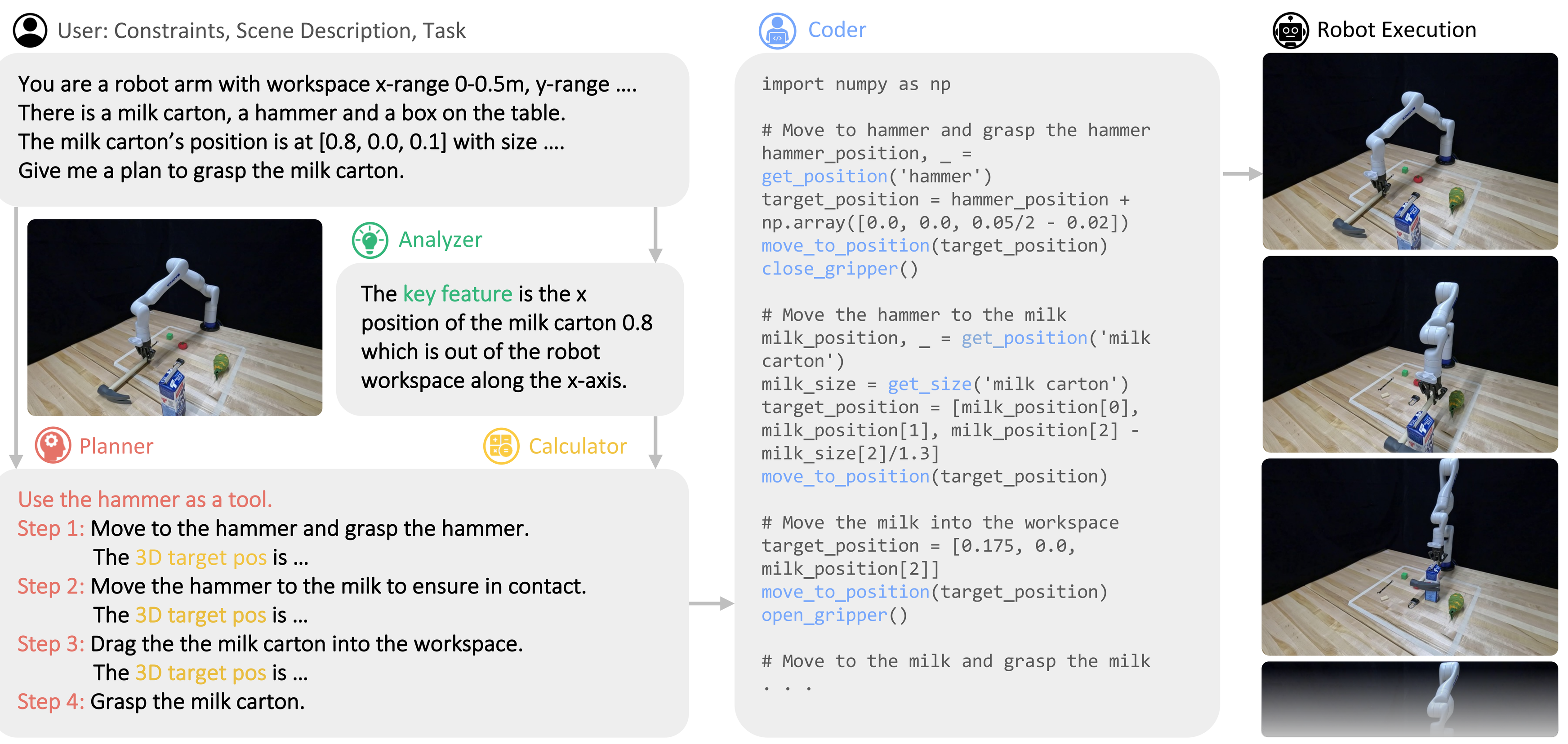
RoboTool has four components:

**Analyzer** for extracting physical concepts affecting task feasibility;

**Planner** for generating high level plan calling parameterized skills;

**Calculator** for calculating the parameters for each skill;

**Coder** to get executable codes.




```
<start of analysis>
The key feature that affects the feasibility
of the plan is the gap between sofa_1 and
sofa_2...

To calculate the gap, we need to consider the
x-axis positions of the two sofas and their
sizes. The center of sofa_1 is at x=0.0 and
its size along the x-axis is 1.5m, so its
edge is at x=0.0+1.5/2=0.75m...

Therefore, the gap between the two sofas is
1.15m - 0.75m = 0.4m, which is larger than
the maximum gap the robot can walk across
(0.1m).
<end of analysis>

<start of description>
The key feature is the gap between sofa_1 and
sofa_2 which is 0.4m, since the robot can
only walk across a gap smaller than 0.1m.
According to the initial configuration, the
constraint is violated initially.
<end of description>
```

Example of Analyzer output.

```
<start of description>
[SOFA_1]: ...
[SOFA_2]: ...
[SURFBOARD]: The surfboard is light
enough for the robot to push and can be
used as a bridge to cross the gap between
the sofas.
[STRIP_OF_CLOTH]: The strip of cloth is
on sofa_1 but it is too thin and small to
be useful in this task.
[Key Feature and constraints]: The key
feature is the gap between sofa_1 and
sofa_2...
[Abstract Plan]: The robot should first
push the surfboard to the edge of
sofa_1...
<end of description>

<start of plan>
- Use the 'get_position' skill to...
- Use the 'push_to_position' skill to
push the surfboard...
...
<end of plan>
```

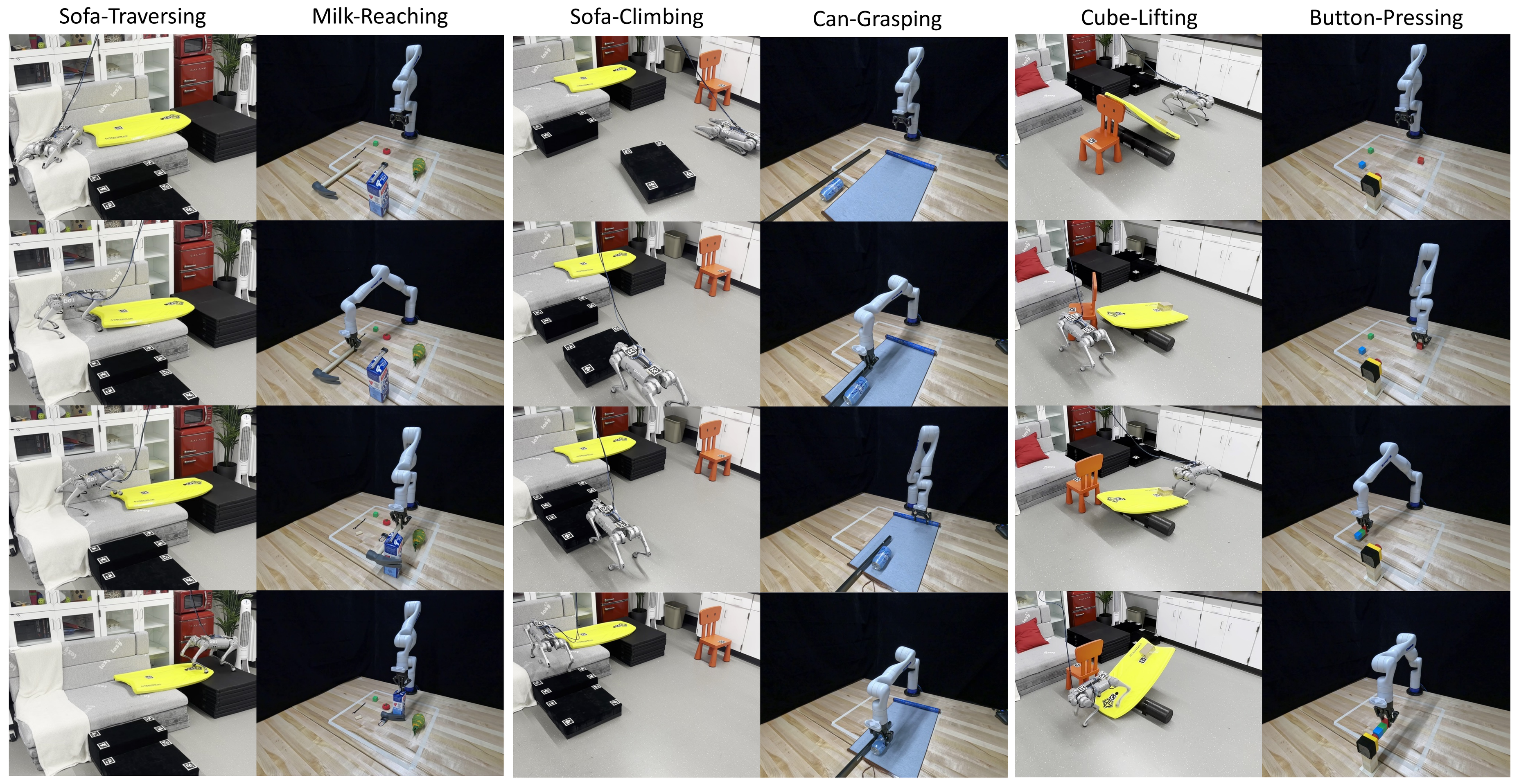
Example of Planner output.

**Takeaway 1:** RoboTool can reason with physical concepts and thus solve physical puzzles, with the help of pretrained LLMs.

**Takeaway 2:** RoboTool can creatively use tools according to their geometric and other physical properties.

**Takeaway 3:** RoboTool can detect activated physical constraints and ground them on the plan, resulting in using tools only when necessary.

## Creative Tool-Use Tasks




We design six creative tool use tasks for two robots.

- Tool selection.
- Sequential tool use.
- Tool manufacturing.

Demos

Project page.

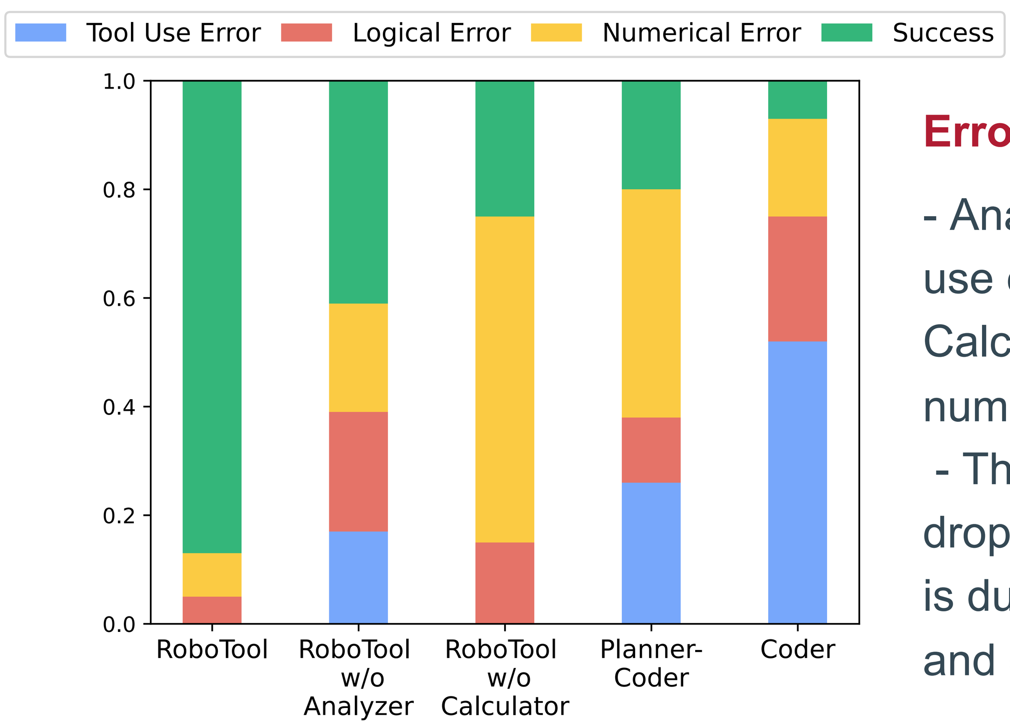
## Experiments in Simulation and Real-world

| | Milk-Reaching | Can-Grasping | Button-Pressing | Sofa-Traversing | Sofa-Climbing | Cube-Lifting | Average |
|---|---|---|---|---|---|---|---|
| **RoboTool** | **0.9** | **0.7** | **0.8** | **1.0** | **1.0** | **0.8** | **0.87** |
| **RoboTool w/o Analyzer** | 0.0 | 0.4 | 0.2 | **1.0** | 0.7 | 0.2 | 0.42 |
| **RoboTool w/o Calculator** | 0.0 | 0.1 | **0.8** | 0.3 | 0.0 | 0.3 | 0.25 |
| **Planner-Coder** | 0.0 | 0.2 | 0.5 | 0.1 | 0.0 | 0.4 | 0.20 |
| **Coder** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.4 | 0.07 |
| **RoboTool (Real World)** | 0.7 | 0.7 | 0.8 | 0.7 | 0.8 | 0.9 | 0.77 |

Success rates of RoboTool and baselines

**Main Results:**

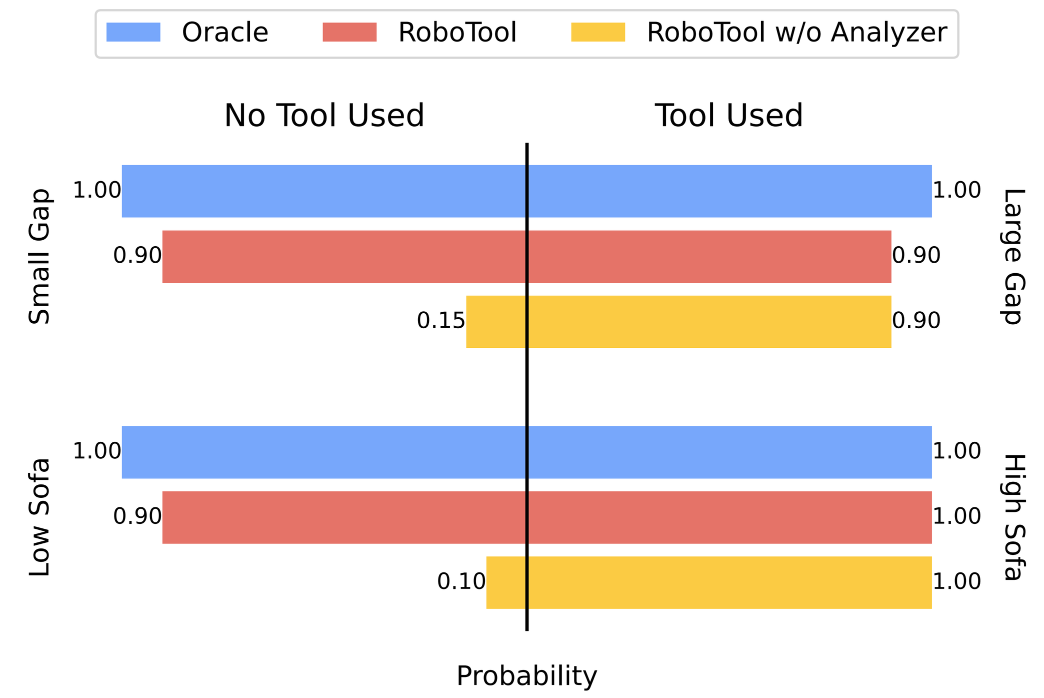
- RoboTool can generate creative tool use behaviors and achieve higher success rates than baselines.




Error breakdown.

**Error Breakdown:**

- Analyer reduces tool use error and Calculator reduces numerical error.
- The performance drop in the real-world is due to perception and execution errors.




Discriminative tool use behaviors.

| | Key Concept and Violated Constraints | Accuracy |
|---|---|---|
| **Milk-Reaching** | Milk's position is out of robot workspace. | 1.0 |
| **Can-Grasping** | Can's position is out of robot workspace. | 1.0 |
| **Button-Pressing** | Button's position is out of robot workspace. | 0.9 |
| **Sofa-Traversing** | Gap's width is out of robot's walking capability. | 1.0 |
| **Sofa-Climbing** | Sofa's height is out of robot's climbing capability. | 0.8 |
| **Cube-Lifting** | Cube's weight is out of robot's pushing capability. | 1.0 |

Key concept accuracy.

**Key concepts and discriminative tool-use behavior:**

- Analyer extracts the key concept, its numerical values and the activated constraints with high accuracy.

- Key concepts help generate discriminative tool use behaviors – using tools only when necessary.