



Views & Comments

通用智能本体

卢策吾^{a,#}, 王世全^{b,#}^a Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China^b Flexiv Ltd., Shanghai 201105, China

1. 引言

在日常的工作和生活中,我们希望智能设备能帮助我们完成各种各样的任务。因此,我们经常需要根据不同的任务和场景设计各种机器人和相应算法。但是,这样远远达不到我们对智能的期望,也满足不了现实中对任务和场景自适应的需求。而反观我们人本身,如果将人类看作一个“智能本体”,用这样一个本体(加上适当工具),我们可以做不同的操作任务适应不同的场景。这也启发我们,我们需要一个通用的智能本体(general-purpose intelligent agent, GIA),大部分任务可以在同一个本体上执行,从而大大提高产业和生活的智能程度。我们还认为,为了接近通用人工智能的目标[1],一个强大的、标准的通用智能本体是必要的。

所以,本文探讨了通用智能本体所需要的体系结构,我们从以下两个角度讨论。类人性(human-like):我们希望通用智能本体能和人类一样有完整的感知能力[2]、决策能力[3],同时能不断接收新知识[4],积累经验,适应新的环境,并且能驱动本体正确实施各种操作,最终完成任务。类计算机性(computer-like):我们希望通用智能本体在体系结构设计上能参照计算机的设计思想[5],比如,我们需要有清晰的模块和模块之间的协议,模块自身可升级,同时用户能通过高级语义编程方式低成本地面向任务描述,而无需关心底层设计。在充分考虑功能的类人性和设计的类计算机性后,我们提出了元操作流模型,提取跨任务间的元操作共性,尽量

用最简单的结构完成尽可能多的任务。本文建议通用智能本体应该有五大基本模块:执行模块、中央GIA处理器、感知模块、知识引擎和任务编译器。在下文中,我们将讨论这些模块如何协同地实现在功能上与人类可类比(具有智能性),也同时在设计思路与计算机可类比(具有可操作性)。我们希望设计的体系机构能实现以下的三大特性。

(1) 可迁移性:在执行大多数任务时,无需更换本体,用户只需提供简单地任务描述,无需重新设计底层通用硬件,协议和操作系统;

(2) 可扩展性:和通用计算机的模块(CPU, RAM)一样,各个基本模块相对独立,且可升级;

(3) 可群智性:面向群体智能,知识可增长,经验可共享。随着通用智能本体群完成任务的增加,知识体系得到扩展,学习新任务所需的时间减少。

我们还进一步讨论了如何量化衡量一个本体的智能程度。我们认为任务迁移能力是衡量一个本体智能的最重要标准,所以,提出了机器人学习成本比(agent-human cost ratio, AHCR)的衡量指标,即对于一项新的任务,一个本体需要人类的多少倍时间学会该任务,这个指标可以很好地衡量一个本体在通用智能方面的综合能力。必须注意的是,本文中通用智能本体的概念不是要苛求完成所有人类的操作任务,而是追求尽可能多的任务能在一个统一的本体上完成。因为根据某些特定场景的需要,特制机器人会有更高的效率,比如扫地机器人。这个正如通用计算机和特殊功能计算机的关系,虽

These authors contributed equally to this work.

然绝大多数人使用通用计算机体系结构，但是某些特殊计算需要使用特殊体系结构的计算机。

文中我们先介绍通用智能本体的基本体系结构和评价指标，以及对智能科学的影响，之后讨论在各个行业中的应用。最后做出总结。

2. 元操作流模型

各种行业、各种环境下有着各种各样需要完成的操作任务。通用智能本体的核心就是要用一个简单模型来统一描述这样近乎无穷数目的操作任务。因此，我们提出了元操作流模型。任何一个人类操作任务都可以分解为多个步骤。比如任务“拧开螺丝”，会分解成为抓起螺丝刀，移动螺丝刀对准螺钉，拧动螺丝刀，取出螺丝刀。

我们沿着这一思路，探讨如何把一个大的操作分解成为多个小操作。为了让一个任务的分解更明确，我们定义“可不再向下分解的通用操作”为元操作，比如抓、拧、插入、揉压等。我们要求这些元操作需要具有足够广泛的迁移共性，这些共性是跨任务、跨操作对象的。

比如，我们在医疗行业扭开一个药瓶盖和在家用行业扭开一个牛奶瓶盖具有很强的共性。而且可预期的是，人类的元操作集合是有限的。因此，在定义好元操作集合后，人类的任何操作任务，都可以解析为一个元操作流，也就是一串元操作序列。接下来，我们更加深入地讨论元操作的完备定义和元操作流的构建。

2.1. 元操作三要素

一个完备的元操作定义，除了元操作类型 P （比如抓、拧、插入），需要考虑操作对象 O 、元操作完成后的最终状态 S 集合。某些操作任务最终完成状态 S 可能有多种可能性，比如敲碎玻璃，最终结果敲碎可能有普通玻璃的碎裂状态或钢化玻璃的颗粒状粉碎状态等。所以我们认为 S 是一个集合，元操作 P 针对操作对象 O 顺利执行后，客观环境状态转变至最终状态集合 S 中的一种。我们简称这几个要素为自适应元操作三要素 $\{P, O, S\}$ 。

当本体获取三要素 $\{P, O, S\}$ 后，本体启动元操作 P 的知识，提取物体 O 的状态（视觉、力觉）与属性，做出自适应操作，指导达到完成状态 S 。这需要依赖一个好的知识引擎，细节会在后面介绍。我们从概念上定义 S 集合，但是实际中，如果 S 集合过于庞大无法枚举，

我们将设计（或者训练）一个判别函数来反馈当前状态是否为完成状态，以及该状态的具体情况。

2.2. 元操作流构建

在元操作集合确定后，如何动态地根据当前环境选择元操作是一个开放的学术问题。我们需要一个元操作选择函数 $T()$ 来决定当前时刻的元操作选择，从而形成元操作流。另外，如何基于通用智能本体的基本架构设计更加智能、更加通用的元操作，也是一个开放的学术问题。这些都需要群体智慧来共同推进。

意义：元操作流模型可以使各种复杂的任务的描述和操作简单化。考虑到人类元操作种类有限，很多复杂的任务描述可以变成元操作的组合问题。而同一个元操作，即使面对不同的物体，也会有很强的共性，这使得我们有可能抽取其中的共性，编辑成元操作知识，实现非常强的复用性和任务可迁移性。

3. 通用智能本体体系结构

如图1所示，我们探讨通用智能本体的基本模块：执行模块、中央GIA处理器、感知模块、知识引擎、任务编译器。这些模块作为一个整体在统一的协议下，完成各种任务，不需要为某个特定任务重新设计新的软硬件体系结构。

接下来我们以元操作流模型为逻辑基础，简单介绍五大基本模块。

执行模块。该部分主要由移动平台与操作平台两大硬件体系组成。

(1) 移动平台。通过移动平台空间上的移动，靠近被操作对象，使得被操作对象处于操作平台的工作空间内。常见的移动平台包括滑轨、轮式、四足、双足等系统，移动局限和灵活性依次降低，控制难度依次提升。如何提升控制稳定性、灵活性及运行效率，仍是机器人领域重要的研究方向。

(2) 操作平台。通过对操作对象施加空间上不同位置、不同维度的施力，使得操作对象产生所预期规划的结果。常见的操作平台包括并行、串联等机构类型的机器臂及其末端执行器，驱动方式包括电驱、气动、液压等驱动类型。如何兼顾操作力的精度和灵活度，操作平台的速度、稳定性、负载和能效，是执行模块进一步进化的关键课题。

感知模块。感知模块接收外界信号，包括采集视觉

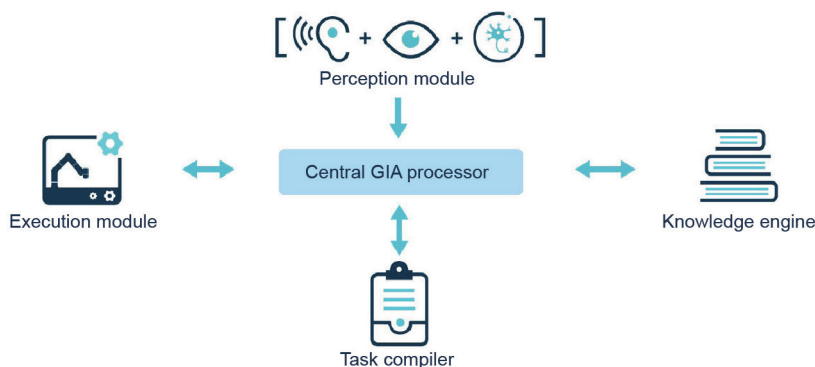


图1. 通用智能本体的五个基本模块。

信号、力觉信号、声音信号等，同时解析环境中的视觉、力觉、听觉等语义。其中，解析视觉语义和听觉语义分别需要使用计算机视觉技术和语音识别。

学术问题：视觉听觉信号采集为相对成熟的技术，其中视觉信息可能包括精确三维信息与热感信息。而目前，基于深度学习的计算机视觉技术对于未在训练集中出现的物体识别能力有限，如何融合环境交互提高对新物体的识别也是一个开放性的学术问题。力觉感知器的反馈信息可能包含低频的六维力觉信号（三维接触力和三维接触扭矩）、高频的触觉信号（接触面的震动状态，可用来辨识接触物体的物理特性和操作状态特性）以及触觉空间信息（力觉信号的空间分布，一般通过阵列形式的传感器获取，类图像特征）。目前，力觉感知器的设计与解析方法在研究上仍处于不断发展的阶段，同时如何更准确全面、成本更低地获取合适的力觉信息并与其他感知维度进行有效融合，是一个开放性的学术课题。

任务编译器。任务编译器是用户向通用智能本体布置任务的统一界面。具体讲，我们提出开发一种“面向任务的高级语言”，用户使用该语言，通过执行模块、感知模块及编译专用设备等媒介进行任务描述，在实际应用中能进行元操作流生成。具体地说，任务编译器负责编译成可以执行的元操作选择函数 $Z()$ 。该函数能根据感知模块所反馈的信息，结合知识引擎提供知识，从有限的元操作库中选择当前最佳的元操作。和传统计算机编程语言不同，用户不止能通过编写程序代码的形式，甚至能通过动作示教、自然语言等多种方式实现任务描述。

学术问题：如何描述一个任务（特别是复杂操作任务）从而生成（或者学习出）一个机器可执行的元操作选择函数 $Z()$ ，也将是一个崭新的科学问题。通过演示学习是一个直接的解决方案，并已研究了很长时间[6]，

但它通常是特定于机器人的，而且成本高昂，无法记录复杂任务的每一个可能的成功路径。考虑到自然语言处理（NLP）技术最近已取得了相当大的进步[7]，通过自然语言向机器人布置任务也是很有前途的方案；然而，自然语言是开放的，而我们更容易从闭合词汇集中学习到技能。

进一步我们要思考机器教学（machine teaching），而不只是机器学习（machine learning）。之前，大家关注机器学习是因为，机器教学问题相对简单，只需要做样本标注就可以，而如何让机器理解任务则是一个更加有难度的问题。

技术展望：对于有固定步骤的任务（比如工业场景），我们可以直接编写元操作顺序。而对于每一步都需要动态决策使用什么样的元操作，目前比较可探索的方法需要观察人类的决策行为，采用模仿学习。这一过程中，“未来预测”，也是一个关键的技术点，人类在选择元操作流不需要大规模试错，而是基于对未来的基本预判，这将大大地提升决策成功的可能性。

知识引擎。在统一的协议下，不同用户可以对物体和物体操作进行知识编辑，对知识库进行扩展。比如，微波炉厂商生产微波炉，可以对其操作知识进行知识编辑，这些知识通过统一格式上传到云端的中央AIG知识引擎。通用智能本体对环境进行解析识别，在知识库中找到对应知识，或者找到最相似的知识集合，对当前案例所需要的操作进行推理。

学术问题：①如何定义物体知识以及物体操作知识的编辑协议；②面对大量动态知识（比如各种元操作知识）如何使用人机交互完成；③当知识库中没有绝对匹配知识时，如何依靠相近知识进行推理。

技术展望：我们希望构建一个全方位的知识库，包括：

（1）基于物体属性的知识引擎：收集大量三维物体

信息，并编辑物体属性，包括物体各个部分的功能、使用方式和物理属性等。目前，没有一个包含所有标注信息的物体数据库。作为当前最大的物体数据集，ShapeNet [8]标注了一些属性，但是对象和属性的覆盖都不完整。然而，如何建立物体属性标注的有效协议也是一件重要的事情。当我们建立了物体属性空间后，对物体的理解会有更强的泛化性。当遇到新的未见物体，虽然没见过，在通过多次交互与识别获得其属性后，我们仍然能较好地理解该物体，并且将这些理解运用到操作上。这样的设置是ShapeNet无法满足的。

(2) 自适应元操作知识引擎：对于每一个特定元操作，在处理不同物体对象时具有一定共性，并且这种共性是跨任务的。我们希望能抽取同一元操作下的共同操作知识，面对不同处理对象能非常好地完成操作任务。具体讲，需要设计（或者学习）一个 M 函数。对于第 i 个操作元，给定操作对象物体集合 O （可能包括对象物体、工具等）、最终状态 S ， M_i 输出一个可执行的指令 u 来完成元操作，数学上可以写为 $u = M_i(O, S)$ 。函数 M 的构建需要结合实时的力觉、视觉等感知信息及引擎中的相关知识。 u 为基于特定控制策略和感知信息的一段机器执行动作。要注意的是，物体属性知识会是操作 O 的主要依据（比如插入前需要判断物体的哪些特征可插、材质属性对插入力的影响等）。操作元的概念有两种：显式的（即基于规则的）[9]和隐式的（即数据驱动的）[10]。传统上，显式方案从未尝试将所有日常操作联合起来，并将注意力集中在几个特定的小任务上，如抓取、移动和清扫[11]。很难将规则定义的原语组合成相当复杂的任务。相反，隐式方案的目标是完全通过数据来学习操作模式。考虑到任务拓扑的复杂性，可能无法收集所需的适当数据量。我们的想法与显式方案密切相关，因为我们为每个原语分配一个人类可识别的名称。

(3) 任务知识引擎：当我们建立和完善元操作知识引擎时，我们需要根据任务确定元操作的组合步骤，其中也存在许多步骤次序不固定的任务。目前，人类的行为被海量的视频所记录，我们可以解析这些行为在各种环境和操作物体下的元操作序列[12]，来获得任务知识引擎[13]。这可以很大程度提高用户使用任务编辑器描述任务的效率。任务解析具有挑战性，因为任务通常可以通过不同的例程来完成。换句话说，常规任务的任务拓扑是复杂的，模型参数无法捕获如此巨大的空间。这就是为什么数据驱动方法对于这样一个问题是不可行的

主要原因[14]。但是，我们的操作元知识引擎将优先处理任务解析问题。

中央GIA处理器。解析任务编译器的任务之后，中央GIA处理器读取感知模块信号，并调用知识引擎中的知识做出综合决策，并向执行模块发出指令。该模块为整个系统的中心计算模块，涉及如何快速提取云端知识引擎，以及实时决策，对计算设计模型提出了一定的要求。

该体系架构有三大特点：

(1) 面向任务语义编辑。用户可以通过任务编译器向通用智能本体发放任务描述。与此配合的，我们提议一套面向任务的高级编程语言，用户无需设计智能本体底层结构。正如我们使用高级语言计算机编程时，不需要设计处理计算机CPU、内存等各种细节。

(2) 标准协议下的可升级模块。感知模块、执行模块和中央GIA处理器都有统一的运行协议，以及各模块之间通讯协议，无论是硬件还是软件都可以独立升级。

(3) 知识与经验可增长。在建立一定物体和操作知识编辑协议后，知识引擎是对全体用户开放的，可以在统一规范下提交物体、元操作、任务等知识模块，实现知识不间断增长。

讨论：我们提出通用智能本体既要从小智能功能上与人类类比，又要从设计上与计算机类比（表1）。

类比人类：中央GIA处理器可类比大脑负责逻辑区域，负责综合处理；感知模块可类比眼睛、耳朵、触觉神经，大脑视觉听觉触觉理解模块获取环境信息；知识引擎可类比大脑记忆区间，执行模块类比人的手和脚；而任务编译器类似大脑中的语言理解区间，用于理解用户任务描述。

类比计算机：任务编译器类似计算机中的编译器，使得各种程序可以执行在中央GIA处理器（类比CPU）上。而知识引擎可以类比存储器。而感知模块自然可以类比计算机的输入设备、键盘、鼠标。但是，计算机没有办法影响外部物理世界，所以执行模块没有对应计算机模块。我们列出类比表格，如表1所示。

4. 标准

我们需要一个可量化指标来衡量智能通用本体的性能。可迁移性是直接衡量一个本体的通用智能程度的标准，即完成一项新任务所需要的学习成本是多少。这也是当前以深度学习为基础的人工智能技术的瓶颈所在

表1 人类功能器官与计算机设计模块类比较表

GIA modules	Comparable human organs	Comparable computer modules
Execution module	Hands, feet	-
Perception module	Eyes, ears, tactile nerves	Keyboard, mouse
Task compiler	Brain language section	Compiler
Knowledge engine	Brain memory section	Memory
Central GIA processor	Brain logic section	CPU

[15]。基于此，我们提出以AHCR作为衡量指标（单位是小时），我们定义如下。

对于某一任务A，我们令：

T_{agent} ：从用户开始编写任务A描述程序起，到本体学会任务A为止所需的平均时间；

T_{human} ：人类学习者从学习任务A起，到学会任务为止所需的平均时间。

因此，我们可以计算任务A的AHCR如下：

$$AHCR = T_{agent} / T_{human}$$

T_{agent} 也称为“本体教学总成本”。这里用“平均时间”是考虑到每个人使用的时间会有所偏差。AHCR的概念是一个本体需要人类的多少倍时间完成一个新任务的学习。我们希望这个数值接近1或者小于1。

这一指标能全面地衡量一个本体的通用智能能力。因为“教学完成时间”涉及两个因素：①任务编程器的成熟程度，属于“教”所需要的时间，如果我们描述一个任务需要大量的时间，比如大量的硬规则（hard-coding）编写时间，那么将会增大“教学总时间”，而最理想的情况，我们能用自然语言或肢体语言描述任务；②本体学习需要的时间，比如模型训练。要特别注意的，预计大部分任务需要多轮人机交互（human in loop），用户要根据任务反馈结果重新编程，这也考验本体与人的交互学习（active learning）效率。另外，当本体的知识引擎越强大和基于知识引擎的推理能力越强大时，所需要的时间也就越少。

任务库（Scalable Task Zoo）。由于用单个任务来衡量一个本体的迁移能力会有一定偏差，我们需要建立一个任务库（Task Zoo）来综合衡量，而且其中的难度和任务种类要有一定多样性和代表性。这样能防止对智能本体的衡量的偏差（bias）。我们将平均Task Zoo上的所有AHCR结果（Average AHCR）来衡量一个智能本体的标准。我们希望的情况是，随着任务数增加，学习总时间会减少。这里将会涉及一个崭新的学术问题，就是在Task Zoo里哪一些任务应该先学，哪一些需要慢学的策略问题。因为人类学习任务也有一个从易到难的过

程，这样的总学习成本最小。

5. 群体智能

我们提出的通用智能本体的进步依赖于群体智能（swarm intelligent）的发展。即我们需要建立可持续的学术与商业社群（community）和有效的组织方式。我们期待以下社群的出现。

知识引擎社群：社群成员提交和审核各种元操作、物体和任务知识。

感知模型社群：社群成员提交和审核各种感知模型，用于比如各种物体检测识别器。

Task Zoo社群：我们希望Task Zoo是可扩展的，所有成员都可以提交新的任务，而我們也需要成立一个审核委员会审核提交的测试任务是否可以加入Task Zoo。测试委员会则是组织对多个本体的测试。

任务编译器社群：我们鼓励多种“高级任务编程语言”共存与竞争，经过一定时间后，出现被用户普遍认可的语言，这也需要一个任务编译器社群的支撑。

6. 对人工智能的影响

我们认为通用智能本体不能简单地看作AI赋能设备。我们相信这样一个本体会大大推进智能科学的发展。目前，主流的深度学习面临的问题是，对于新的任务，没有训练过的数据性能会大大下降，换句话说就是可迁移性较弱。通用智能本体的提出可以很大程度上提高机器对任务的可迁移性。

对环境的深刻理解：智能体需要对其所在环境（特别是操作对象）有深度理解。基于有泛化能力的元操作集合，通过与物体交互操作，智能体对物理世界产生更加深刻的理解，能更好地表征物体各种概念，比如，物体功能、操作方式、材质等。人类也正是通过与物体的交互，实现对物体属性层面的深刻理解。比如，我们理解剪刀这个概念，我们通过交互判断它是否能剪断物

体。如图2所示，即使训练数据中没有这样的图片，也能判断这是剪刀。而传统深度学习算法（如果没有事先训练过类似图片）无法识别。

通用智能本体提供了和世界交互的基础，可以获得更多的物体属性概念，这将有别于传统的深度学习。基于理解物体属性，系统会对未训练的物体也能很好地理解。从某种程度讲，借助元操作构建，本体实现了从模式识别到物体理解，对于没有见过的物体有很好的泛化能力。

可增长群体智能：虽然群体智能被提出来已经有20年的历史，但是要跨任务比较困难[16]。而通用智能本体的推出提供了一个很好的机会。因为我们的逻辑基础是元操作，而元操作在不同任务下具有很强共性。随着大量操作被（通用智能本体）群体各个成员所创造和记录，这种跨任务共性会被提取出来进一步支持任务可迁移。另外，通用智能本体有统一的协议，当人类社会广泛使用同一个通用智能本体，经验共享变得更加容易，对未知任务的迁移能力越来越强。再次，每个元动作具有很强的通用性和层级性，能够对海量的智能空间进行更加高效的结构化索引，为多维远距的共性语义创建合理链接。

7. 通用智能本体的应用

通用智能本体将会提高各个领域的效率，预期会在工业制造、医疗、餐饮、农业、家用等很多领域产生巨大影响。比如，在家用领域，我们不可能对每个家庭特殊环境的每个任务重新设计硬件、软件或学习模型，这也是阻碍家用机器发展的重要原因。考虑到各个领域的难易程度，通用智能本体在实际应用中的产业落地在时间上会有先后，我们可以粗略分成四个程度（Level）（表2）。

我们认为操作步骤未知，每次要决策使用什么元操作，对智能程度要求较高，比如修理物品等。而操作环境和操作物体在知识引擎的帮助下，即使有所变化也能相对简单地进行自适应和处理。所以，我们可以大致给

出各个行业在各Level上的分布（图3）。每个行业很难说绝对处于哪个Level上，而是跨Level的。其中，工业通用智能本体是相对简单的阶段，因为任务确定后，大部分操作属于操作对象与环境变化不大、操作步骤已知的情况。到了家用通用智能本体，需要和人（一个变化很大的物体）交互，而且步骤也是不确定的（比如照顾老人）。

所以，我们建议第一步先完成工业通用智能本体，因为这是近期可实现的，也可以非常好地形成一个通用智能本体的基本雏形。后面各个行业的智能本体可以在这个雏形上升级，最终实现跨行业的通用智能本体。

执行模块。由于大多数集中于装配的工人都固定在前方，因此我们提出的工业通用智能本体的执行模块可以是一对机器人手臂，其7个自由度与人类的7个自由度相匹配。在装配复杂的零件时，人类通常依靠触觉和力来感知是否装配成功，这是一个在零件上施加不同维度的力和力矩（共6个自由度）的过程。因此，手臂需要能够灵活地对零件施加不同的力和力矩组合，这意味着机器人手臂也应该具有较高的6个自由度的控制性能。

感知模块。要有一个对力具有高控制性能的手臂，一个高质量的力/力矩传感器是必需的。它可以嵌入手臂关节或末端执行器。除了触觉，还需要视觉来识别物体，定位每个物体的位置和方向，以及在每个物体上装配特征以支持复杂的手眼协调。可考虑采用RGB-D深度相机来实现三维视觉。



图2. 我们可以识别传统深度学习算法无法识别的没有事先训练过的图片。

表2 通用智能本体的四级应用难度

Level	Condition of object and environment	Process step
Level 1	Both the environment and the objects of manipulation vary slightly	Known process step
Level 2	Either the manipulation environment or the objects of manipulation varies greatly	Known process step
Level 3	Both the environment and the object of manipulation vary greatly	Known process step
Level 4	Both the environment and the object of manipulation vary greatly	Unknown process step

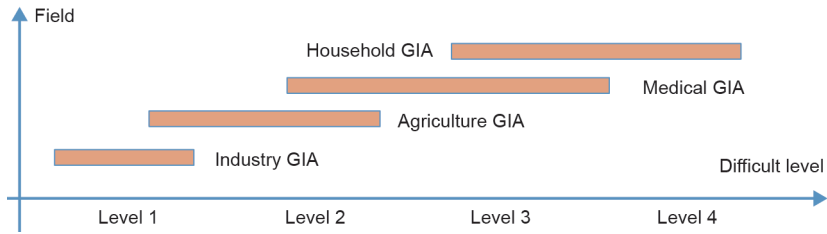


图3. 各产业通用智能本体在不同难易程度的分布。

任务编译器。在生产线上，编制完善的标准操作规程（SOP）和作业指导书对工人执行任务是非常必要的。因此，最直接的任务编译器是能够解析标准操作规程并将其转换为正确的元操作序列，并从标准操作规程中的图片中学习对象，从而链接每个元操作。与通用智能本体在其他领域的应用相比，工业通用智能本体的元操作集相对较小，因此另一个更简单的编译器可以是字段工程编辑元操作流的编程板。

知识引擎。我们在这里提出了一组常规组装过程所需的元操作，这些元操作可能并不是所有组装任务的完整集合。

(1) 插入、夹紧A部件并将其外/内特性滑动到B部件的内/外特性上。可能的特征包括连接器、轴/孔、螺钉/孔。

(2) 垫料，将A部件压在B部件的上面，在A部件的表面来回移动，使具有紧密公差A和B可以配合在一起。可能的特征包括使彼此配合过渡或冲压配合的零件几何形状。

(3) 将A部件拧紧、车削、安装到B部件，特征包括螺纹、管螺纹等。

工业通用智能本体。与实现各个领域广泛应用所需的通用智能本体能力相比，通用智能本体应用于工业自动化是在一个更结构化的环境中运行，具有更明确的过程，因此实现起来没有那么困难。

如今，经过了几十年的技术发展，各种类型的机器人和定制的设备已经在现代化的生产线上得到了有效的利用。成熟的自动化工作包括焊接、喷漆、装料等。然而，有一些程序需要更复杂的材料处理，去适应零件的符合性和组件的非完美扭矩控制，如复杂的装配，这些程序仍然只能依靠人类完成。我们提出了一种工业通用智能本体的可能形式，能够进行上述类型的工作。

8. 总结

本文我们提出了一种新的概念，通用智能本体，即追求各种任务间的通用可转移性，并提出了通用智能本体的核心元操作流模型和体系结构。我们相信如果通用智能本体的概念能实现，不仅将极大地改进人类生活和生产，而且会促进智能科学的发展。

References

- [1] Shapiro SC, editor. Encyclopedia of artificial intelligence. 2nd ed. New York: John Wiley; 1992.
- [2] Wiss JF, Parmelee RA. Human perception of transient vibrations. J Struct Div 1974;100(4):773–87.
- [3] Odum HT. Environmental accounting: energy and environmental decision making. New York: Wiley; 1996.
- [4] Studer R, Benjamins VR, Fensel D. Knowledge engineering: principles and methods. Data Knowl Eng 1998;25(1–2):161–97.
- [5] Hayes JP. Computer architecture and organization. New York: McGraw-Hill; 2002.
- [6] Mandlekar A, Zhu Y, Garg A, Booher J, Spero M, Tung A, et al. RoboTurk: a crowdsourcing platform for robotic skill learning through imitation. In: Proceedings of the 2nd Conference on Robot Learning; 2018 Oct 29–31; Zürich, Switzerland; 2018. p. 879–93.
- [7] Devlin J, Chang MW, Lee K, Toutanova K. Bert: pre-training of deep bidirectional transformers for language understanding. 2018. arXiv:1810.04805.
- [8] Chang AX, Funkhouser T, Guibas L, Hanrahan P, Huang Q, Li Z, et al. ShapeNet: an information-rich 3D model repository. 2015. arXiv:1512.03012.
- [9] Morales A, Prats M, Sanz P, Pobil AP. An experiment in the use of manipulation primitives and tactile perception for reactive grasping. In: Proceedings of the 2007 Robotics: Science and Systems; 2007 Jun 27–30; Atlanta, GA, USA; 2007.
- [10] Dasari S, Ebert F, Tian S, Nair S, Bucher B, Schmeckpeper K, et al. RoboNet: large-scale multi-robot learning. In: Proceedings of the 3rd Conference on Robot Learning; 2019 Oct 30–Nov 1; Osaka, Japan; 2019.
- [11] Fang K, Zhu Y, Garg A, Kurenkoy A, Mehta V, Li FF, et al. Learning task-oriented grasping for tool manipulation from simulated self-supervision. 2018. arXiv:1806.09266.
- [12] Li YL, Xu L, Liu X, Huang X, Xu Y, Chen M, et al. HAKE: human activity knowledge engine. 2019. arXiv:1904.06539.
- [13] Li YL, Zhou S, Huang X, Xu L, Ma Z, Fang HS, et al. Transferable interactiveness knowledge for human-object interaction detection. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition; 2019 Jun 16–20; Long Beach, CA, USA; 2019.
- [14] Xu D, Nair S, Zhu Y, Gao J, Garg A, Li FF, et al. Neural task programming: learning to generalize across hierarchical tasks. In: Proceedings of the 2018 IEEE International Conference on Robotics and Automation; 2018 May 21–25; Brisbane, QLD, Australia; 2018.
- [15] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.
- [16] Kennedy J. Swarm intelligence. In: Zomaya AY, editor. Handbook of nature-inspired and innovative computing. Boston: Springer; 2006. p. 187–219.