Supplementary Materials

S1 Materials and Methods

S1.1 Episodic energy and time requirements

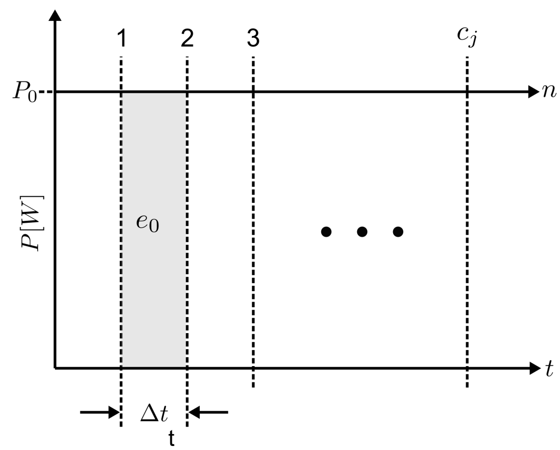

Fig. S1. Power consumption per episode.

Energy requirements. Under Asm. **??**, the energy consumption of the n-th episode $e_j(n)$ is the constant product

$$e_j(n) = \underbrace{P_0 \Delta t}_{\text{constant}} = e_0. \tag{1}$$

Consequently, the energy consumed by a robot learning a skill s_j is directly proportional to the skill complexity c_j; i.e.

$$E_j = \sum_{n=1}^{c_j} e_j(n) = e_0 c_j. \tag{2}$$

The energy spent on learning $\mathcal{S}$ under the absence of knowledge transfer is

$$E_{\mathcal{S}} = \sum_{j=1}^{N_{\mathcal{S}}} E_j = e_0 \sum_{j=1}^{N_{\mathcal{S}}} c_j \tag{3}$$

Time requirement. Similarly, the total learning time $T_{\mathcal{S}}$ for a simple agent is

$$T_{\mathcal{S}} = \Delta t \sum_{j=1}^{N_{\mathcal{S}}} c_j. \tag{4}$$

S1.2 The different learning paradigms

Isolated Learning (IsL) A robot performs IsL when it learns each skill in $\mathcal{Z}_k$ one after another from the ground up, disregarding the accumulating knowledge from already learned skills. In such a case the rate of convergence and the initial remaining knowledge for all skills are given by

$$f_{j,k}(\cdot) = -\alpha \tag{5a}$$

$$g_{j,k}(\cdot) = 1, \tag{5b}$$

where $\alpha > 0$ models the rate at which a robot in isolation learns any given skill. Relying on Asm. **??** we can assign a value to α by using the fundamental complexity c_0 as follows

$$\alpha = -\frac{1}{c_0}\log(\epsilon). \tag{6}$$

Since in isolated learning $c_{j,k}^{(IsL)} = c_0$, the trial episodes required by one single robot to learn the skills in the cluster $\mathcal{Z}_k$ is given by

$$C_k^{(IsL)} = \sum_{j=1}^{N_{\mathcal{Z}}} c_{j,k}^{(IsL)} = N_{\mathcal{Z}} c_{j,k}^{(IsL)} = N_{\mathcal{Z}} c_0 \tag{7}$$

Similarly, the total trial episodes to learn the universe of skills is simply

$$C_{\mathcal{S}}^{(IsL)} = N_{\mathcal{K}} N_{\mathcal{Z}} c_0. \tag{8}$$

Multi agent case. Suppose that a batch of m robots is used to learn the same number of skills in parallel in a given cluster $\mathcal{Z}_k$. Such a strategy only distributes equally the total number of episodes by the number of available robots; i.e.

$$^{||}C_k^{(IsL)} = \overbrace{\frac{1}{m} C_k^{(IsL)}}^{\text{episodes per robot}}. \tag{9}$$

Incremental Learning (IL) It corresponds to the continuous aggregation and exchange of knowledge from *intra-cluster* skills. Referring back to Asm. **??**, the knowledge from skills belonging to a cluster $\mathcal{Z}_k$ can be be leveraged by an agent in virtue of their significant similarity. As depicted in Fig. **??**, a robot (r_1 in this case) learns every skill in $\mathcal{Z}_1$ with a rate α —the self loops— but also retains and uses the acquired knowledge to learn subsequent skills. The effect of incremental learning on the knowledge collection rate can be modeled to be directly proportional to the number of learned skills as

$$f_{j,k}\left(N_{\zeta_k}\right) = -\alpha\left(\eta N_{\zeta_k} + 1\right), \tag{10}$$

where $\eta > 0$ represents the efficiency of knowledge exchange from ζ_k to $s_{j,k}$. Different potential models might be used to model the depletion of the initial remaining knowledge represented by $g_{j,k}\left(N_{\zeta_k}\right)$, e.g. a linear decay rate, our expectation is that, under the assumption that a learning strategy involving the ordering of skills according to similarity and their balanced distribution in the different clusters, $g_{j,k}\left(N_{\zeta_k}\right)$ might naturally resemble an exponential decay that is strongly dependent on N_{ζ_k}. Such considerations motivate our choice of the following function

$$g_{j,k}\left(N_{\zeta_k}\right) = e^{-\delta N_{\zeta_k}}, \tag{11}$$

again with a factor $\delta > 0$ controlling the rate at which the exponential converges. Similar to α, using Asm. **??** δ can be defined as

$$\delta = -\frac{1}{N_{\mathcal{Z}}}\log(\epsilon). \tag{12}$$

Essentially, such choice of δ implies that the remaining knowledge in a cluster after seeing all its skills is negligible. Via the exchange factors (η, δ), in incremental learning the knowledge about every new skill gets gradually increased by leveraging previous knowledge, resulting in

$$\bar{\sigma}_{i,j}^{(IL)}(n) = e^{-\alpha\left(\eta N_{\zeta_k}+1\right)n} e^{-\delta N_{\zeta_k}}.$$

As the complexity $c_{j,k}$ of a skill can also be interpreted as the number of trial episodes required for the remaining knowledge to go below a threshold ϵ; i.e.

$$\bar{\sigma}_{i,j}^{(IL)}(n)\Big|_{n \geq c_{j,k}^{(IL)}} \leq \epsilon.$$

Then, under this scheme the complexity $c_{j,k}^{(IL)}$ to learn a new is skill in the cluster results in

$$c_{j,k}^{(IL)} = -\frac{\log(\epsilon) - \log\left(\bar{\sigma}_{j,k}^{(IL)}(0)\right)}{\alpha(\eta N_{\zeta_k} + 1)} = -\frac{\log(\epsilon) + \delta N_{\zeta_k}}{\alpha(\eta N_{\zeta_k} + 1)}. \tag{13}$$

The total number of trial episodes C_k that an agent following an incremental learning strategy needs to learn the $N_{\mathcal{Z}_k}$ skills in a cluster $\mathcal{Z}_k$ is given by

$$C_k^{(IL)} = \sum_{j=1}^{N_{\mathcal{Z}}} c_{j,k}^{(IL)}. \tag{14}$$

Multi-agent case. If m robots are used in parallel to divide the load of learning the tasks then

$$^{||}C_k^{(IL)} = \sum_{j=1}^{\frac{N_{\mathcal{Z}}}{m}} c_{j,k}^{(IL)}. \tag{15}$$

In essence, using m robots without exchanging knowledge only subdivides the learning in every cluster into m smaller problems *without adding any additional benefit to the rate at which knowledge is acquired.*

Transfer + Incremental Learning (TIL) Transfer learning (TL) alone refers to the one-time *inter-cluster* exchange of knowledge. Considering $\mathcal{K} = \{\mathcal{Z}_k\}_{k=1}^{N_{\mathcal{K}}}$ to be the set of all available skill clusters, TL represents the exchange of knowledge from the skills learned in different *origin* clusters $\mathcal{O} = \{\mathcal{Z}_1, \mathcal{Z}_2, \ldots, \mathcal{Z}_{k-1}\}$ to the skills that will be learned in a *destination* cluster $\mathcal{Z}_k$ (see Fig. **??**). Concretely, the effect that TL has on the skills of the destination cluster is the reduction of the initial remaining knowledge and the increase of the initial learning rate for all the skills in the k-th cluster via the parameter β_k; i.e.

$$f_{j,k}\left(N_{\zeta_k}\right) = -\alpha\left(\frac{\eta N_{\zeta_k}+1}{1-\beta_k}\right), \tag{16}$$

and

$$g_{j,k}\left(N_{\zeta_k}\right) = (1-\beta_k)\,e^{-\delta N_{\zeta_k}}. \tag{17}$$

In essence, β_k is the head start granted by knowledge transfer from other clusters to the skills in $\mathcal{Z}_k$. We argue that $0 < \beta_k < 1$ since it represents the *aggregated* knowledge exchange factor from the different origin clusters $\mathcal{Z}_c$ to the target cluster $\mathcal{Z}_k$. Let $0 < \beta_c < 1$ be the transfer contribution factor of a single origin cluster $\mathcal{Z}_c$. Additionally, consider that

$$\sum_{c=1}^{N_{\mathcal{K}}} \beta_c \leq 1, \tag{18}$$

as 1 represents all the knowledge in $\mathcal{S}$. Asm. **??** implies that β_c is equal for all the clusters. In this work we select $\beta_c = 1/N_{\mathcal{K}}$ for simplicity. The aggregated transfer factor β_k is the sum of the individual factors from the already-visited clusters; i.e.

$$\beta_k = (k-1)\,\beta_c = (k-1)\,\frac{1}{N_{\mathcal{K}}}. \tag{19}$$

Consequently, the remaining knowledge when transfer and incremental learning are used in conjunction is

$$\bar{\sigma}^{(TIL)}_{j,k}(n) = \left(1-\beta_k\right) e^{-\alpha\left(\frac{\eta N_{\zeta_k}+1}{1-\beta_k}\right)n}\, e^{-\delta N_{\zeta_k}}. \tag{20}$$

Similar to incremental learning, the complexity to learn a skill in transfer learning is

$$c^{(TIL)}_{j,k} = -\frac{1-\beta_k}{\alpha(\eta N_{\zeta_k}+1)}\left[\log(\epsilon) + \delta N_{\zeta_k} - \log(1-\beta_k)\right] \tag{21}$$

and the total number of episodes C_k that an agent requires to learn the $N_{\mathcal{Z}_k}$ skills is merely their sum

$$C^{(TIL)}_{k} = \sum_{j=1}^{N_{\mathcal{Z}}} c^{(TIL)}_{j,k}. \tag{22}$$

Multi-agent case. If m robots are used in parallel to divide the load of learning the tasks then, the transfer of knowledge from cluster to cluster is also divided by the number of robots, this implies that (19) changes to

$$^{||}\beta_k = \frac{1}{m}\beta_k. \tag{23}$$

Correspondingly, when using transfer learning in parallel β_k is replaced by $^{||}\beta_k$ in (21). Then, similar to IL, the total number of episodes to learn the skills in a cluster is

$$^{||}C_k^{(TIL)} = \sum_{j=1}^{\frac{N_{\mathcal{Z}}}{m}} c_{j,k}^{(TIL)}. \tag{24}$$

This case is depicted on Fig. **??**, where two robots r_1 and r_2 learn skills in four different clusters. The shaded areas are the subclusters of skills learned by each robot. Since they do not share knowledge between them, each robot has access only to the knowledge it has collected and cannot benefit from one another.

S2 Use case results

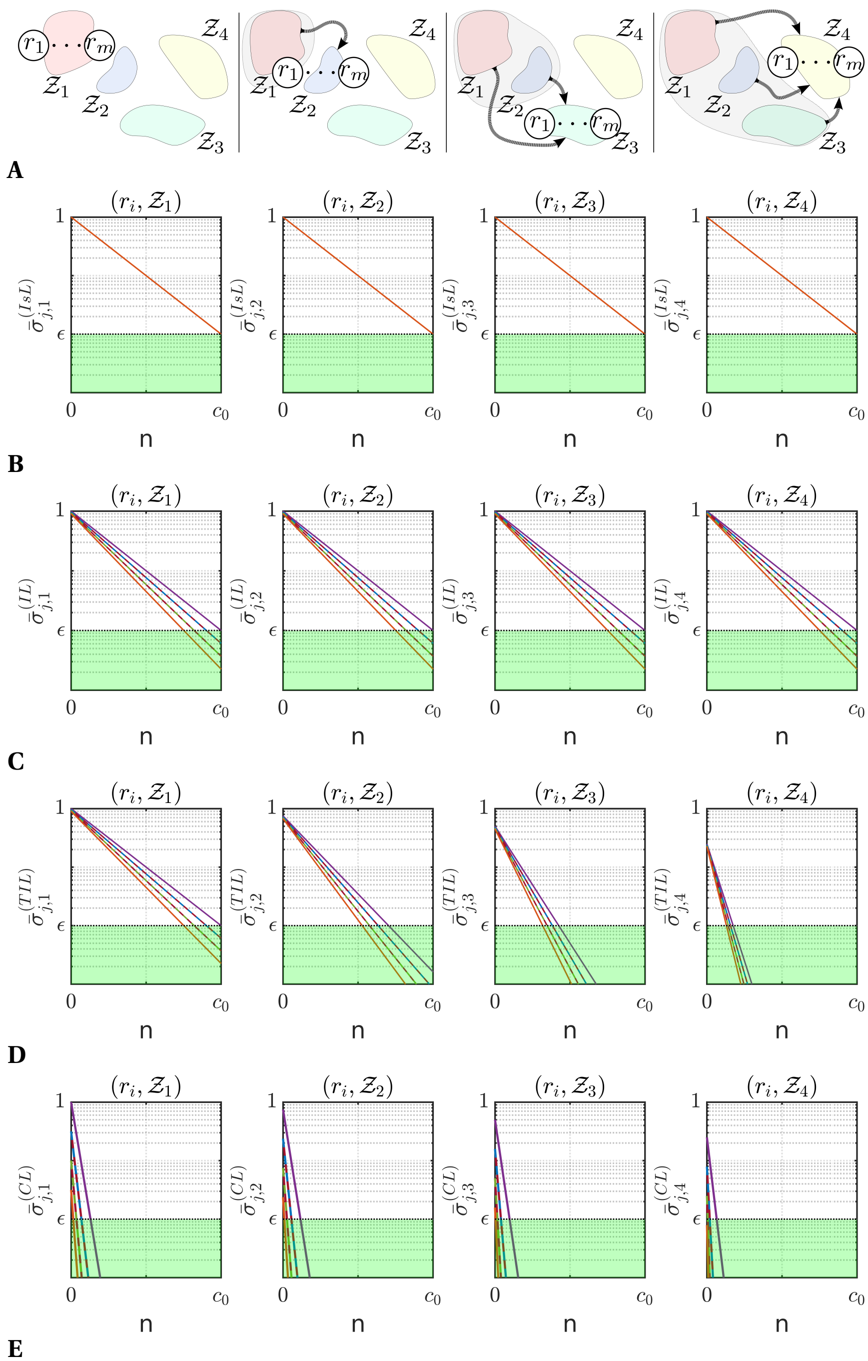

Fig. S2. Scenario 1: **A** the skills of each cluster are learned by the m robots in succession, **B** isolated learning, **C** incremental learning, **D** incremental + transfer learning, **E** collective learning.

S3 Supporting statistics

S3.1 Industrial and cobot statistics

According to the International Federation of robotics the unit sales of collaborative robots in relation to conventional industrial robots has been constantly increasing in the past years (**?**). Recent data, see Fig. S3, shows that cobots now make almost 15 % of the sales.

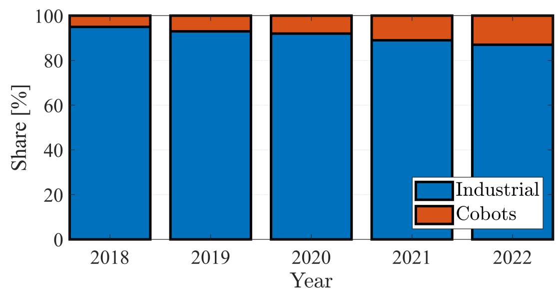

Fig. S3. Unit sales share industrial robots to cobots

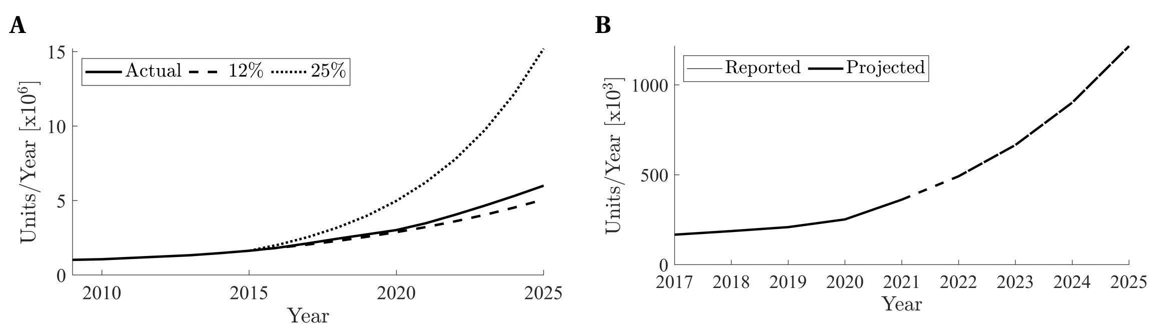

Fig. S4. Forecasts for robot operational stock. **A** industrial robots install base forecast and **B** cobots install base forecast.

S4 Industrial robot energy consumption support data

S4.1 Industrial robots

According to (?) and available press releases of different robotic companies (?, ?, ?), the approximate distribution of the industrial robot install base per manufacturer is shown in Fig. S5**A**.

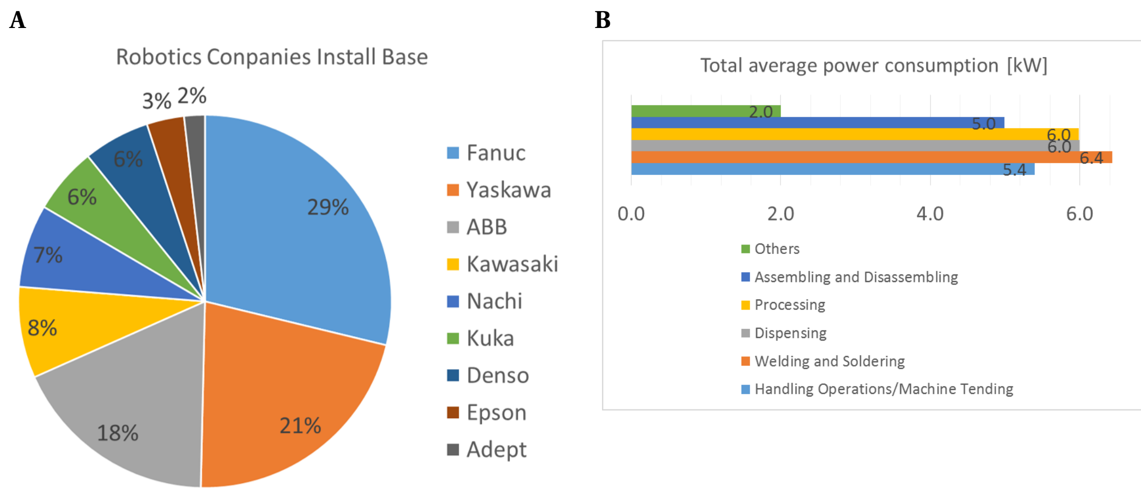

Fig. S5. Industrial robots statistics. **A** Percentage of installed industrial robots per manufacturer and **B** average power consumption of industrial robots per category.

Since Fanuc, Yaskawa, and ABB make for two-thirds of the total install base of industrial robots, we took the power consumption of the robots from those manufacturers to estimate the total power consumption. After surveying the datasheets for their robot portfolio, the average power consumption for each model was estimated. Additionally, every manufacturer classifies their robots according to one or more possible applications, which can be grouped into the application categories defined by the IFR. The average power consumption was calculated for every application using the values reported in the robot datasheets. Finally, the power consumption for each category was computed as a weighted average based on the companies' market share percentage (assuming that 68 % is the total number of robots)[1].

[1]These numbers should be used with discretion since there is no available information on which are the

The estimated power consumption per robot application is shown in Fig. S5**B**. Using these numbers and the estimated operational stock of industrial robots reported in (*?*) and by the International Federation of Robotics (see Fig. S4**A**), the estimated worldwide industrial robot energy consumption was computed and shown in Fig. **??**.

S4.2 Collaborative robots

To approximate the energy consumption of cobots we looked at the power consumption per payload of various manufacturers, see Fig. S6 resulting in an average power consumption of approximately 40 W. Together with a typical power consumption of the robot controller of 60 W (*?*), we consider a total of 100 W power demand. Similar to the industrial robots, the worldwide energy consumption was calculated assuming a 24/7 operation.

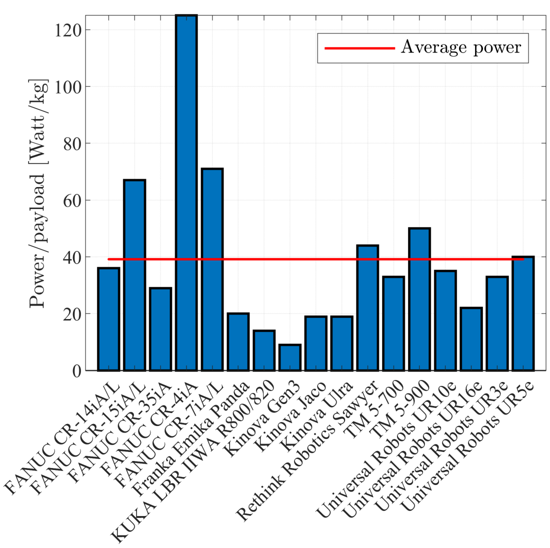

Fig. S6. Power consumption per payload for different cobots.

most common installed robot models. This information may change the estimation.