2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

SARA-RT: Scaling up Robotics Transformers with Self-Adaptive Robust Attention

Isabel Leal*, Krzysztof Choromanski*, Deepali Jain*, Avinava Dubey*, Jake Varley*,
Michael Ryoo, Yao Lu, Frederick Liu, Vikas Sindhwani, Quan Vuong, Tamas Sarlos, Ken Oslund,
Karol Hausman, Kanishka Rao
isabelleal@, kchoro@, jaindeepali@, avinavadubey@, jakevarley@, mryoo@, yaolug@, frederickliu@, sindhwani@, quanhovuong@,
stamas@, kenoslund@, karolhausman@, kanishkarao@; google.com
Google
1600 Amphitheatre Parkway Mountain View, CA 94043, *equal contribution

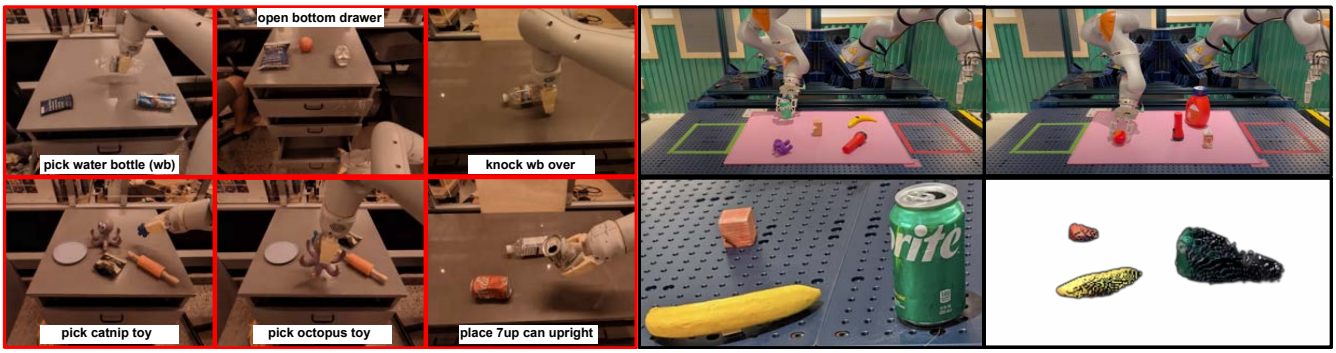

Fig. 1: Robotics Transformer policies obtained via *Self-Adaptive Robust Attention* (SARA) in action for three different modalities: vision, language and point clouds and varying sequence lengths (from $\sim$200 to few thousand). **Left:** A 5B vision-language-action model from the RT-2 class [1] (sequence length: $L = 196$). The manipulation policy is conditioned on the text instruction. **Right:** A Point Cloud Transformer (PCT) ([2]) for grasping (point cloud sizes $L \in [800, 4000]$) and an exemplary PC input. Both policies use linear attention up-trained with SARA and provide computational speedups, while maintaining high quality.

***Abstract*— We present Self-Adaptive Robust Attention for Robotics Transformers (SARA-RT): a new paradigm for addressing the emerging challenge of scaling up Robotics Transformers (RT) for on-robot deployment. SARA-RT relies on the new method of fine-tuning proposed by us, called *up-training*. It converts pre-trained or already fine-tuned Transformer-based robotic policies of quadratic time complexity (including massive billion-parameter vision-language-action models or VLAs), into their efficient linear-attention counterparts maintaining high quality. We demonstrate the effectiveness of SARA-RT by speeding up: (a) the class of recently introduced RT-2 models [1], the first VLA robotic policies pre-trained on internet-scale data, as well as (b) Point Cloud Transformer (PCT) robotic policies operating on large point clouds. We complement our results with the rigorous mathematical analysis providing deeper insight into the phenomenon of SARA.**

I. INTRODUCTION & RELATED WORK

The unprecedented semantic reasoning capabilities offered by Transformers ([3], [4], [5], [6], [7], [8], [9], [10]) for Robotics gave rise to the new field of machine learning, exploring Transformer-based models, often pre-trained on the internet-scale data, for robotic controllers. Those controllers enable abstract reasoning from the multi-modal input and have already led to several recent breakthroughs in Robotics, including: high-level planning with large language models (LLMs) ([11], [12], [13], [14]), policies transforming natural language commands into on-robot executable code ([15], [12]), multi-modal sensor fusion [16], finally the first vision-language-action robotic manipulation powered by massive vision language models [1]. Interestingly, even when not fine-tuned, Transformer models trained on massive web corpus seem to learn structural reinforcement learning priors that can be leveraged to conduct trajectory optimization for certain control tasks via in-context learning ([17], [18]) with carefully designed prompts, as very recently shown in [19].

Is the Transformer-driven revolution in Robotics a straightforward path to AGI and all that is left is to train larger and deeper models, pre-trained on even more massive datasets ? One of the challenges that is still not addressed, yet is of critical practical importance in Robotics, is a prohibitively expensive space and time complexity of the aforementioned models. For example, the 35M-size parameter RT-1 model [20], providing dramatic generalization improvements over non-Transformer approaches and marking the birth of the class of Robotics Transformer architectures (RTs), even though of modest size, already operates with the frequency at most 3Hz. The problem only deepens for larger 1B+ parameter models such as RT-2 [1].

In this work we make a step towards solving it. We present *Self-Adaptive Robust Attention for Robotics Transformers* (SARA-RT): a new paradigm for addressing the emerging challenge of scaling up Robotics Transformers for on-robot

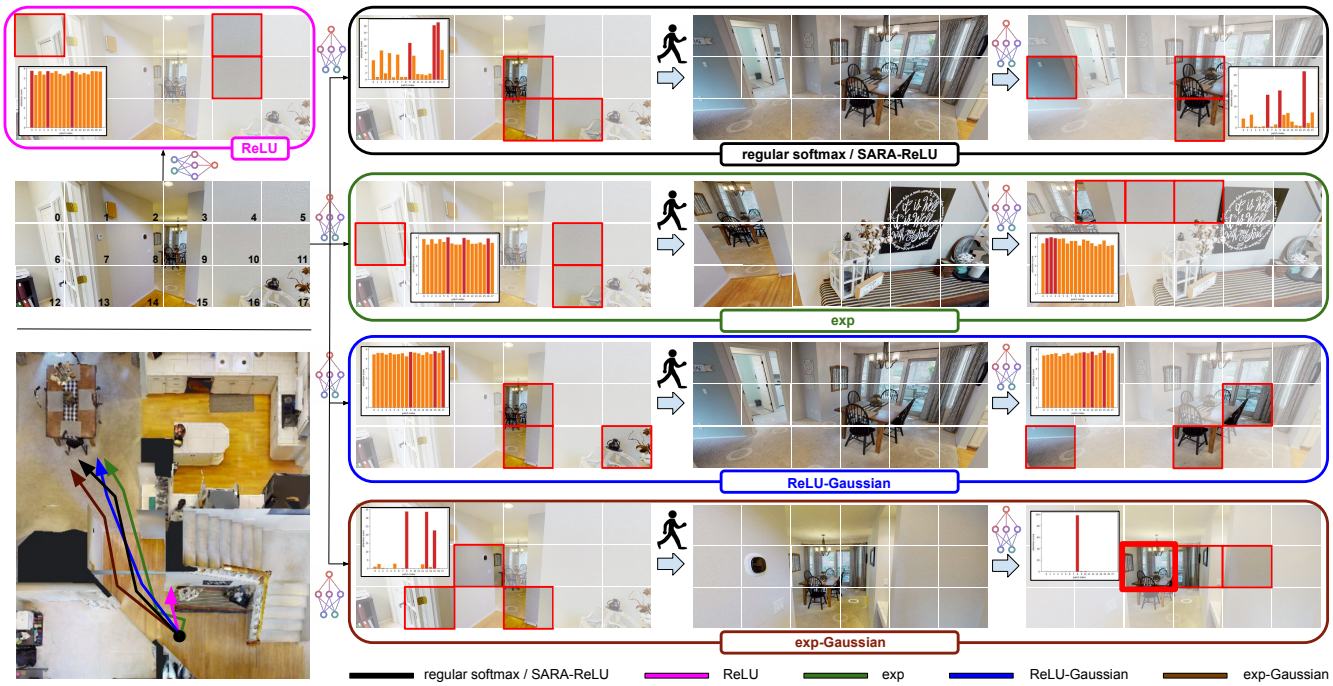

Fig. 2: VR navigation via VL attention models on Matterport environments ([21]). The top-down view of the scene is in the lower-left corner. The agent's initial view with indexed image patches is right above. For each view, we highlight three patches with highest attention scores and present the distribution of the attention scores over all the patches. At any given point of time, the agent chooses one of them with probability proportional to the score. The initial parts of the executed trajectories are visualized. The trajectories are also included in the top-down view. The agent is given a text command: *"Go to the table"* (or the corresponding image). The ReLU-variant (pink block) directly leads to the collision with the wall. For the exp-variant the agent can navigate, but is temporarily distracted by the unrelated object in the lower-right corner of the initial view. Both variants with Gaussian-matrix pre-processing (blue and brown blocks) apply $m = 2048$ (patch/text embedding dimensionality for CLIP with ViT-B is $d = 512$) and lead to efficient navigation. The attention scores distribution for the ReLU-Gaussian remains flat though. The SARA variant with $m = d$, and where matrix $\mathbf{G}$ is trained to mimic regular softmax-attention for every new view, produces indistinguishable (spiky) attention scores and leads to efficient navigation (black block). In practical applications involving Transformers, SARA-training is conducted once in the so-called *up-training* process (see: Sec. II-C).

deployment. SARA-RT relies on the new method of fine-tuning proposed by us, called *up-training*. It leverages it to convert pre-trained or already fine-tuned Transformer-based robotic policies of quadratic space and time complexity (including massive billion-parameter vision-language-action models or VLAs), into their efficient linear-attention counterparts maintaining high quality. We demonstrate the effectiveness of SARA-RT by speeding up: (a) the class of the aforementioned RT-2 models, the first VLA robotic policies pre-trained on internet-scale data, as well as (b) Point Cloud Transformer (PCT) [2] robotic policies operating on large point clouds (see: Fig. 1). We complement our results with the rigorous mathematical analysis explaining the phenomenon of the effectiveness of SARA-RT's.

II. SELF-ADAPTIVE ROBUST ATTENTION VIA THE KERNELIZATION VIEWPOINT

A. Developing intuition: zero-shot navigation via VL models

Consider a vision-based VR navigation agent, conditioned on the images of the target objects: $t_1, ..., t_M$ or the corresponding natural language commands (e.g. *"Go to the table"*). For simplicity, we will assume that the targets do not need to be visited in any particular order. First we show that vision-language (VL) models can be used in a zero-shot manner for steering the agent. We take the VR environments from the Matterport ([21]) real estate virtual tour sites.

We consider a purely zero-shot attention-based control mechanism, where the action $\mathbf{a}_i$ of the agent corresponding to the particular target t_i $(i = 1, ..., M)$ is defined as follows:

$$\begin{cases} \mathbf{a}_i = \sum_{j=1}^{N} s(i,j)\bar{\mathbf{a}}_j, \\ s(i,j) = \frac{\mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)}{\sum_{l=1}^{N} \mathrm{K}(\mathbf{q}_i, \mathbf{k}_l)} \end{cases} \tag{1}$$

and: **(1)** N stands for the number of the patches of the *patchified* version of the input image, **(2)** $\{\mathbf{k}_i\}_{i=1,...,N} \subseteq \mathbb{R}^{d_{QK}}$ is the set of their latent embeddings (keys), **(3)** $\mathbf{q}_i \in \mathbb{R}^{d_{QK}}$ is the embedding of the target image (or the corresponding text command) (query), **(4)** $\mathrm{K} : \mathbb{R}^{d_{QK}} \times \mathbb{R}^{d_{QK}} \rightarrow \mathbb{R}$ is the kernel function (similarity measure) defining how the agent *attends* and **(5)** $\bar{\mathbf{a}}_j$ are *base actions* (e.g. left/right turns, going forward/back). We call the coefficients $s(i,j)$ the *attention scores*. We inherit the embeddings from the pre-trained CLIP model [7] with the ViT-B vision backend.

CLIP model as a zero-shot navigation agent is presented in Fig. 2 (top black block). As a baseline, we applied the *softmax-kernel* $\mathrm{K}(\mathbf{x}, \mathbf{y}) \stackrel{\mathrm{def}}{=} \exp(\mathbf{x}^\top \mathbf{y})$, a default choice for attention. To visualize the main concepts, we interpret $\mathbf{a}_i$ as the expected base action over discrete probabilistic distribution with probabilities $(s(i,j))_{j=1,...,N}$ and sample

a base action from its variant truncated to the three top-score actions. The base action corresponds to "clicking" on the random pixel of the corresponding patch. As we see, the baseline agent can reach the target without any additional task-specific fine-tuning of the CLIP embeddings.

Now consider the hypothetical case when the set of targets is large, even comparable with N. In that case, quadratic space and time complexity $O(MN)$ might become prohibitively expensive. This can be addressed algorithmically if K admits a bi-linearization, i.e. can be re-written as a linear (dot-product) kernel in the new input-space:

$$\mathrm{K}(\mathbf{x}, \mathbf{y}) = \mathbb{E}[\phi(\mathbf{x})^\top \phi(\mathbf{y})] \tag{2}$$

for some (randomized) map: $\phi : \mathbb{R}^{d_{QK}} \rightarrow \mathbb{R}^m$. In that case, each $\mathbf{a}_i$ can be approximated as:

$$\begin{cases} \tilde{\mathbf{a}}_i = \frac{\Psi \phi(\mathbf{q}_i)}{\Gamma \phi(\mathbf{q}_i)}, \\ \Psi = \sum_{j=1}^{N} \bar{\mathbf{a}}_j \phi^\top(\mathbf{k}_j), \\ \Gamma = \sum_{j=1}^{N} \phi^\top(\mathbf{k}_j) \end{cases} \tag{3}$$

and thus all $\tilde{\mathbf{a}}_i$ can be computed in time & space $O(M+N)$ (by computing Ψ and Γ first). Linear attention mechanisms is an area of active research ([22], [23], [24], [25], [26], [27], [28], [29], [30]), yet most of the theoretical focus is on developing low-variance variants unbiasedly approximating original softmax-kernel attention (known as particularly expressive due to its combinatorial nature with spiking attention patterns). Those apply random ϕ via random Gaussian projections. This computational overhead makes them practically attractive only for M, N large enough (usually 4K+). Besides, linear attention usually produces some performance gap as compared to its brute-force softmax counterpart.

On the other side of the spectrum are linear attention models with $\phi = \phi_f$ given as: $\phi_f(\mathbf{z}) = (f(z_1), ..., f(z_{d_{QK}}))^\top$ for some $f : \mathbb{R} \rightarrow \mathbb{R}$, where f is an easy to compute function, such as: ReLU or exp. Those are in practice very fast and provide computational gains for M, N as small as 128, but are well known for being less accurate than the variants providing unbiased estimation. For that reason, they were not of much theoretical interests.

At first glance, we confirm these findings for the zero-shot VL navigation. The results for ϕ_f with $f = \mathrm{ReLU}$ and $f = \mathrm{exp}$ are presented in Fig. 2 in the pink- and green-border box respectively. The ReLU-variant leads to the agent hitting the wall and the exp-variant, even though reaches the target, is distracted from it at some point, attending to the unrelated object. Finally, none of them provides distinctive attention spikes, characteristic for the regular softmax-variant (sharply attending to the most relevant image regions), but a relatively flat distribution of the attention scores.

B. Improving ϕ_f for zero-shot navigation via randomization

There exists however a very simple trick that improves ϕ_f, making our agent a more efficient zero-shot navigator. It suffices to preprocess the input to ϕ_f by the random matrix. To be more specific, we define the randomized version of ϕ_f as: $\phi_f^{\mathrm{rand}}(\mathbf{z}) = f(\mathbf{G}\mathbf{z})$, where $\mathbf{G} \in \mathbb{R}^{m \times d_{QK}}$ is a Gaussian matrix with entries taken i.i.d from $\mathcal{N}(0, 1)$ (sampled once and used for all inputs $\mathbf{z}$) and f is applied element-wise.

As we show in Fig. 2 (blue- and brown-border boxes), this modification enables both the ReLU and exp variants to reach their targets with no distractions and furthermore already leads to spiking attention patterns for the exp variant. This will become more clear later. In Sec. III, we show that $\phi_{\mathrm{exp}}^{\mathrm{random}}$ is intrinsically related to the unbiased estimators of the original softmax-attention.

C. The birth of SARA: learnable pre-processing

Randomized mappings ϕ_f^{rand} in Fig. 2 apply $m = 2048$ and thus, as we discussed before, may not be relevant for tasks with $M, N < 1K$. The core idea behind *Self-Adaptive Robust Attention* is that rather than being Gaussian, matrix $\mathbf{G}$ can be learned. We define SARA mapping ϕ_f^{SARA}, acting on the raw d-dimensional embeddings: $(\mathbf{x}_i)_{i=1}^M$, $(\mathbf{y}_j)_{j=1}^N$ of objects/tokens (rather than queries/keys $(\mathbf{q}_i)_{i=1}^M / (\mathbf{k}_j)_{j=1}^N$ that for instance for Transformers are obtained from the former by linear projections W_Q and W_K) as follows for learnable $\mathbf{v} \in \mathbb{R}^m$, $\mathbf{G}_Q, \mathbf{G}_K \in \mathbb{R}^{m \times d}$ and a *Hadamard product* $\odot$:

$$\phi_{f,1}^{\mathrm{SARA}}(\mathbf{z}) = \mathbf{v} \odot f(\mathbf{G}_Q \mathbf{z}), \ \phi_{f,2}^{\mathrm{SARA}}(\mathbf{z}) = \mathbf{v} \odot f(\mathbf{G}_K \mathbf{z}) \tag{4}$$

The kernel values $\mathrm{K}(\mathbf{x}_i, \mathbf{y}_j)$ (for: $\mathrm{K} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$) are then expressed as dot-products $\phi_{f,1}^{\mathrm{SARA}}(\mathbf{x}_i)^\top \phi_{f,2}^{\mathrm{SARA}}(\mathbf{y}_j)$, as in Eq. 2. Instead of one mapping ϕ, now there are two: ϕ_1, ϕ_2, but we have perfect equivalence since in Eq. 2, $\mathbf{x}/\mathbf{y}$ corresponds to the query/key $\mathbf{q}_i/\mathbf{k}_j$ obtained by two different mapping from $\mathbf{x}_i/\mathbf{y}_j$. Furthermore, the analogous computational complexity analysis follows.

It turns out that then one can take $m = d_{QK}$, making SARA a practical mechanism even for $M, N < 1K$. And indeed, even though $M, N > 1K$ for our point clouds empirical results in Sec. IV-A, the speed improvements for the RT-2 from Sec. IV-B were obtained for $M, N < 200$. As showed in Fig. 2, with learnable $\mathbf{G}$ and $m = d_{QK}$ (here $d = d_{QK}$), one can learn linear attention mechanisms that for all practical purposes produce indistinguishable attention patterns as compared with the regular softmax-attention, thus leading to the same navigation trajectories for our VR tasks.

The VL navigation is a convenient "macroscopic" case study, but our main targets are Transformer-architectures for Robotics, where queries for the whole images or text instructions are replaced by their counterparts for image-patches, text-tokens or even individual point of the point clouds (PCs), produced by the Transformer-encoded policies.

We thus propose the process of the self-adaptation of their attention modules, that we refer to as *up-training*, which can be implemented as replacing regular softmax-attention with its ϕ_f-encoded variants and fine-tuning them on data from the downstream robotic tasks. Learnable pre-processing corresponds here to fine-tuning matrices W_Q and W_K ([3]) from Transformers' attention modules, but in the linear attention context.

III. THE MATHEMATICS OF SARA-RTs

As a warm-up, we show that a linear attention mechanism using $\phi_{\mathrm{exp}}^{\mathrm{random}} : \mathbb{R}^{d_{QK}} \rightarrow \mathbb{R}^m$ leads to the unbiased

estimation of the softmax-kernel up to the fixed constant if the inputs to the softmax-kernel have fixed length (e.g. CLIP embeddings that are by default L_2-normalized).

Lemma 3.1: The following holds for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{d_{QK}}$ with $\|\mathbf{x}\| = \|\mathbf{y}\| = r$ and softmax-kernel $\mathrm{K} : \mathbb{R}^{d_{QK}} \times \mathbb{R}^{d_{QK}} \rightarrow \mathbb{R}$:

$$m \cdot \exp(r^2) \cdot \mathrm{K}(\mathbf{x}, \mathbf{y}) = \mathbb{E}[(\phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{x}))^\top \phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{y})] \quad (5)$$

Proof: Note that the *positive random feature* vector $\phi^+(\mathbf{z})$ from [22] is of the following form for any $\mathbf{z} \in d_{QK}$:

$$\phi^+(\mathbf{z}) = \frac{1}{\sqrt{m}} \exp(-\frac{\|\mathbf{z}\|^2}{2}) \exp(\mathbf{G}\mathbf{z}) \quad (6)$$

By Lemma 1 from [22]: $\mathrm{K}(\mathbf{x}, \mathbf{y}) = \mathbb{E}[\phi^+(\mathbf{x})^\top \phi^+(\mathbf{y})]$. That implies Eq. 5 and completes the proof. ■

One can also derive corresponding concentration results.

Lemma 3.2: Given the conditions from Lemma 3.1, $g_{r,t}^m = \frac{t}{\sqrt{m}} \exp(r^2(2\cos(\theta) \ + \ 1))\sqrt{(1 - \exp(-2r^2(1 + \cos(\theta))))}$ for an angle θ between $\mathbf{x}, \mathbf{y}$, the following holds for $t > 0$:

$$\mathbb{P}[|\frac{(\phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{x}))^\top \phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{y})}{m \exp(r^2)} - \mathrm{K}(\mathbf{x}, \mathbf{y})| > g_{r,t}^m] \leq \frac{1}{t^2} \quad (7)$$

Proof: By Lemma 2 from [22], the variance of the estimator $\widehat{\mathrm{K}}(\mathbf{x}, \mathbf{y}) = \phi^+(\mathbf{x})^\top \phi^+(\mathbf{y})$ satisfies: $\mathrm{Var}(\widehat{\mathrm{K}}(\mathbf{x}, \mathbf{y})) = \frac{1}{m} \exp(-(\|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2))(\exp(2\|\mathbf{z}\|_2^2) - \exp(\|\mathbf{z}\|_2^2))$ for $\mathbf{z} = \mathbf{x} + \mathbf{y}$. To complete the proof, it suffices to note that $\widehat{\mathrm{K}}(\mathbf{x}, \mathbf{y}) = \frac{(\phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{x}))^\top \phi_{\mathrm{exp}}^{\mathrm{random}}(\mathbf{y})}{m \exp(r^2)}$, $\|\mathbf{z}\|_2^2 = 2r^2(1 + \cos(\theta))$ and apply Chebyshev's Inequality [31]. ■

Our main result shows that for the appropriate $\mathbf{v}, \mathbf{G}, f$, mappings $\phi_{f,u}^{\mathrm{SARA}}$ $(u = 1, 2)$ lead to the accurate approximation of the attention score matrix $\mathbf{A} = [\mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)]_{i=1,...,M}^{j=1,...,N} \in \mathbb{R}^{M \times N}$ in the Transformer, importantly with the number of trainable parameters at most **logarithmic** in $\mathbf{A}$ size.

Theorem 3.3: Consider normalized Transformer's attention layer with queries $(\mathbf{q}_i)_{i=1}^M$ and keys $(\mathbf{k}_j)_{j=1}^N$ of length r. Denote: $\tau = \min_{i,j} \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)$, $\rho = \max_{i,j} \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)$.

Take $m = \lceil \frac{2\rho^2}{\delta^2 \tau^2} \log(2MN) \exp(-\frac{r^2}{A}) \rceil + 1$ for $A < 0$, $\delta > 0$. Then there exist $\mathbf{v} \in \mathbb{R}^m, \mathbf{G}_1, \mathbf{G}_2 \in \mathbb{R}^{m \times d}, f : \mathbb{R} \rightarrow \mathbb{R}$ such that the approximate attention matrix $\widehat{\mathbf{A}}$ (implicitly) given by the mappings $\phi_{f,u}^{\mathrm{SARA}}$ satisfies:

$$\|\mathbf{A} - \widehat{\mathbf{A}}\|_\infty \leq \delta \quad (8)$$

Proof: Take $f = \exp$. We will show that if:

$$\mathbf{G}_1 = \sqrt{1 - 4A}\mathbf{G}\mathbf{W}_Q, \ \mathbf{G}_2 = \sqrt{1 - 4A}\mathbf{G}\mathbf{W}_K,$$
$$\mathbf{v} = (1 - 4A)^{\frac{d_{QK}}{4}} (\exp(A\|\mathbf{g}_1\|^2), ..., \exp(A\|\mathbf{g}_m\|^2))^\top, \quad (9)$$

where $\mathbf{G} \in \mathbb{R}^{m \times d_{QK}}$ is a Gaussian matrix with iid entries from $\mathcal{N}(0, 1)$ and $\mathbf{g}_1, ..., \mathbf{g}_m$ are its rows, then the resulting $\widehat{\mathbf{A}}$ satisfies the condition with nonzero probability for m given in the statement. Here $\mathbf{W}_Q$ and $\mathbf{W}_K$ are standard linear projections of the raw embeddings $(\mathbf{x}_i)_{i=1}^M$ and $(\mathbf{y}_j)_{j=1}^N$, defining queries $(\mathbf{q}_i)_{i=1}^M$ and keys $(\mathbf{k}_j)_{j=1}^N$ and we assume that attention matrix entries $\tilde{\mathbf{A}}_{i,j}$ before row-normalization satisfy: $\tilde{\mathbf{A}}_{i,j} = \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j) = \exp(\mathbf{q}_i^\top \mathbf{k}_j)$ (softmax attention).

Take $\epsilon = \frac{\delta}{\rho}$. Denote by p_ϵ the probability of an event $E(\epsilon) = \{\exists_{i,j} |\widehat{\mathrm{K}}(\mathbf{q}_i, \mathbf{k}_j) - \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)| > \epsilon \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)\}$. Note that by Theorem 3.1 from [28], $\frac{\phi_{f,1}^{\mathrm{SARA}}(\mathbf{x}_i)^\top \phi_{f,2}^{\mathrm{SARA}}(\mathbf{y}_j)}{m \exp(r^2)}$ is an unbiased estimator of $\mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)$. Furthermore, by Theorem 4.2. from [28], the probability that $|\frac{\phi_{f,1}^{\mathrm{SARA}}(\mathbf{x}_i)^\top \phi_{f,2}^{\mathrm{SARA}}(\mathbf{y}_j)}{m \exp(r^2)} - \mathrm{K}(\mathbf{q}_i, \mathbf{k}_j)| > \tau\epsilon$ is at most $r_{\tau\epsilon} = 2\exp(-\frac{m\tau^2\epsilon^2}{2} \exp(\frac{r^2}{A}))$. Thus, by the union bound, we get: $p_\epsilon \leq MNr_{\tau\epsilon}$. Note that if $E(\epsilon)$ does not hold, then $|\mathbf{A}_{i,j} - \widehat{\mathbf{A}}_{i,j}| \leq \epsilon\rho$ for every i, j. That completes the proof, since for m defined in the statement of the theorem, $p_\epsilon < 1$. ■

IV. EXPERIMENTS

A. Robotic Point Cloud Transformers

In our first set of experiments, we trained robotic grasping Transformer policies operating on the point cloud (PC) data. Since Point Cloud Transformers ([2]) usually use relatively long 1K+ sequences, even for simple objects, the unscalability of the brute-force quadratic attention is a severe problem.

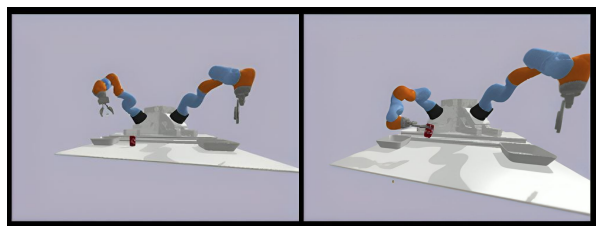

Fig. 3: The simulator used to train PC-input grasping policies and the successful coke can grasp with corresponding reward $r = 1$.

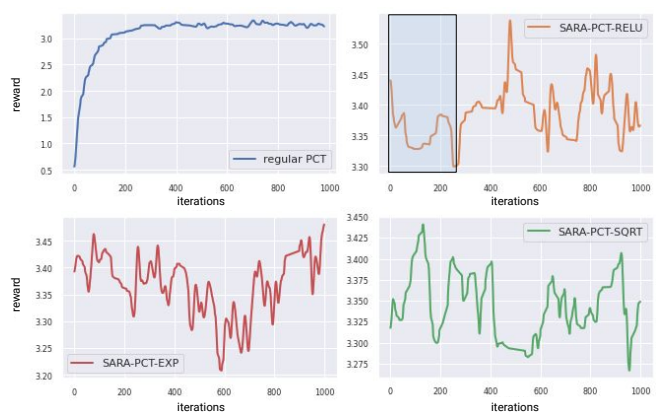

Fig. 4: Training regular PCT policy as well as three variants of SARA with $f \in \{\mathrm{ReLU}, \exp, \mathrm{sqrt}\}$ (up-training from the regular PCT checkpoint). The adaptation process of the linear attention for the ReLU variant is highlighted. For each of the five objects to grasp, an agent receives a binary reward (1: grasp, 0: no grasp). The reported reward is computed as their average over 100 trials. For all the variants, all the checkpoints (even in the preliminary stage of up-training) provide good quality policies (the y-axis for SARA-plots is the magnified version of the short high-reward interval of the y-axis for the regular PCT training.)

1) The setting: A grasping policy receives a PC from a Realsense camera. A pass-through filter removes all points except for those of table top objects. These points are hierarchically clustered into objects. A single object PC is then passed into a PCT policy to produce a grasp pose offset from the center of the object point cloud. The observation space has 3 components: **(1)** *cloud*: $N \times 3$ point cloud with the workspace origin at the mean of the object's cloud; **(2)** *center*: (x, y, z) cloud center in workspace frame of

reference; **(3)** $major\_axis$: (x, y, z) representing the major axis of the points in the object frame. The grasp pose action is represented by the fingertip position relative to the object center, the approach direction vector in the workspace frame, and the robot arm wrist roll angle. The grasp is then executed open-loop with a Kuka IIWA robotic arm and Weiss gripper.

PCT policy training is conducted in the simulator (Fig. 3) via blackbox optimization ([32], [33], [34], [35]). The BGS variant ([34]) with $l = 50$ perturbation-directions, Gaussian smoothing parameter $\sigma = 0.02$, step size $\eta = 0.02$, $\tau = 30\%$ top directions is applied. In training, an agent sees only $k = 5$ different objects: a coke can, a water bottle, a chalkboard eraser, a banana and an octopus soft toy. Policies learned in sim were ready for the on-robot fine-tuning (also with blackbox methods), but it was not necessary (no major sim-to-real gap was observed). Objects encountered in testing included several of shapes never seen in training.

Fig. 5: **Upper row:** The AB-test setup. Different configurations can vary by the number of objects of the table and their shapes. **Lower rows:** One of the grasps performed by SARA-ReLU in the low-reachability-area setting (target object close to the robotic arm base), requiring careful PC-input-based control of the robotic arm.

2) SARA-PCT vs regular PCT: We trained in sim regular PCT policy till convergence for $s = 1000$ iterations. In the SARA setting, we then conducted up-training with $f \in \{\exp, \mathrm{ReLU}, \mathrm{sqrt} : x \rightarrow x^2\}$, the all-one vector $\mathbf{v}$ and $m = d$. The results are presented in Fig. 4. We see almost immediate adaptation of the linear attention for all SARA variants. Thus we chose (here and for the RT-2 experiments) the simplest ReLU (that can be thought of as the tamed version of the exp variant), on-robot deployed it and compared with the regular PCT in the AB-test. The test consisted of $N = 200$ random object configurations (see: Fig. 5), where for each configurations either SARA-PCT or regular PCT policy is randomly selected. The agent gets a binary reward $r \in \{0, 1\}$ for each grasp (success or failure). The average reward obtained by the regular PCT agent is: $r_{\mathrm{ave}}^{\mathrm{reg}} = \mathbf{0.64}$ and by the SARA-PCT agent: $r_{\mathrm{ave}}^{\mathrm{SARA}} = \mathbf{0.75}$.

We then run speed tests for the regular PCT encoder and SARA-PCT encoder for input sequences of different lengths. Both encoders consist of $t = 2$ Transformer layers with 16-dimensional latent embeddings. The actual sequence length for the on-robot deployment varies from scene to scene, but can easily exceed 1K. As showed in Fig. 6, SARA-PCT provides significant speedups, guaranteeing practically constant inference $\approx 100\mathrm{ms}$ (regardless of the point cloud size), with the attention module not being a computational bottleneck anymore. On the contrary, for the regular PCT the attention module remains a bottleneck and leads to the substantial slowdowns even for relatively small point clouds.

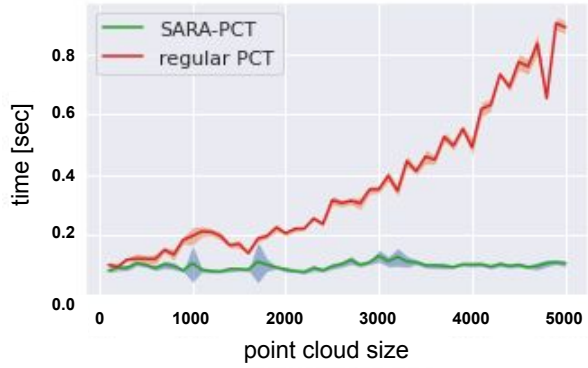

Fig. 6: Speed tests for SARA-PCT and regular PCT. Reported are mean inference times (averaged over $l = 10$ random seeds) for PCT encoders (as well as the corresponding standard deviations; see: shaded regions) as functions of the point cloud size.

B. RT-2 vision-language-action models

1) The setting: Next we consider the class of RT-2 architectures from [1]. Those apply PaLI-X [36]) vision-language-model (VLM) backbones to encode policies taking vision input and conditioned on natural language instructions. We focus on the 5B PaLI-X variant, as more practical for the on-robot deployment than the 55B variant. As described in [1], the action space consists of 6-DoF positional and rotational displacement of the robot end-effector, as well as the level of extension of the robot gripper and a special discrete command for terminating the episode, which should be triggered by the policy to signal successful completion. The VLM takes a text instruction and an image (or a history of images) and produces a sequence of text tokens that can be then converted to actions (see: Sec. IV-B.2 for details).

The pre-trained VLM is fine-tuned on the robotic data from [20]. It consists of expert demonstrations collected with a mobile manipulation robot. Each demonstration is mapped to the natural language instruction from one of the following classes: *"pick object"*, *"knock object over"*, *"open/close drawer"* , *"place object into receptacle"*, *"place object upright"*, *"move object near object"*, *"pick object from receptacle and place on the counter"* (details in [20]).

For SARA variants (with $f = \mathrm{ReLU}$ and all-one vector $\mathbf{v}$), up-training is conducted after the fine-tuning phase. There are two natural places to inject linear attention in PaLI-X models: the ViT encoder and the fuser, where tokens from all the frames as well as text ones are concatenated (self-

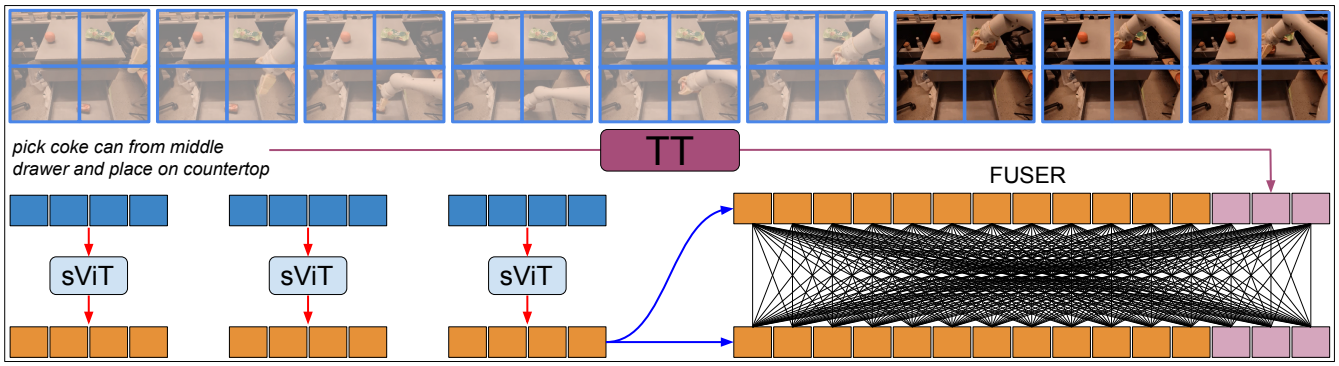

Fig. 7: The scheme of some of the key elements of the PaLI-X backbone of RT-2 from the computational viewpoint, accompanied with the real robot performing text instruction using SARA model. This example applies three-frame history with each frame partitioned into four patches (in practice the number of patches is much larger). Frames are encoded via SARA variants of the ViTs (sViT). Text instruction is separately pre-processed by the text Transformer (TT). In the fuser, all resulting embeddings are concatenated and interacting with each other via self-attention. This self-attention block is yet another good candidate for injecting SARA variants. We leave it to future work.

TABLE I: Comparison of different variants of the RT-2 models with the PaLI-5B VLM backbones. The comparison is conducted on the six regular manipulation task and an additional task measuring generalization level (diverse pick). The mean accuracies for all three models are: **65.8%**, **65.1%** and **76.4%** (up-down). The first two models differ only by the applied attention module (regular or linear).

	pick	**knock**	**open/close drawer**	**drawer place**	**upright**	**move**	**diverse pick**
RT-2 (PaLI-5B), no history + action tokens	81%	86%	67%	39%	57%	98%	33%
SARA-RT-2 (PaLI-5B), no history + action tokens	83%	91%	78%	31%	46%	79%	48%
SARA-RT-2 (PaLI-5B), history + vector representation	100%	91%	89%	56%	51%	81%	67%
number of trials per eval	36	35	18	36	35	48	21

attention modules in both) (see: Fig. 7). In this work, we chose the former, leaving testing the latter to future work.

2) Action representation: Each action can be represented as a vector consisting of: positional and rotational displacements, a scalar tracking gripper's extension level and a sub-vector encoding episode termination command. Traditionally (see: [1] for details), each continuous dimension is quantized into one of the 256 bins, leading to an action given by a sequence of integers, tokenized by PaLI-X (the so-called *action tokens*). We propose an alternative (*vector representation* of the action space), where each continuous dimension is rounded up to four decimal places, leading to the string that can be text-tokenized as usually. We noticed that the vector representation of actions results in higher quality models.

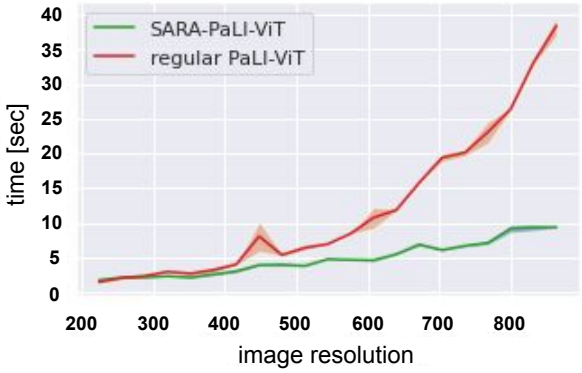

Fig. 8: Speed tests (on a CPU). Reported numbers are as in Fig. 7, but for PaLI-ViT encoders as functions of the resolution of the image ($x_{\text{value}} \times x_{\text{value}}$) for the default 16×16 patch size.

3) SARA-RT-2 vs regular RT-2: We start by comparing RT-2 variant using one frame (no history) and old tokenizer with its SARA counterpart. Both use $L = 196$ vision tokens. As we show in Table I (first two rows), SARA variant maintains good performance across all seven tested tasks (providing better performance on four of them). Both variants have very similar mean accuracy, though SARA variant generalizes better. Besides, regular RT-2 controller needs **53.2** ms (TPU) for the forward pass, while SARA's: **45.7** ms (TPU) (**14%** speedup). Finally, we combine SARA with the new tokenizer from IV-B.2 and the history of $H = 3$ frames. The latter two techniques provide farther accuracy boost, but are more demanding computationally, and thus challenging for direct on-robot deployment. It turns out that the resulting ViT-linear-attention hybrid RT-2 variant (third row in Table I) provides **12%+** mean accuracy improvement, excelling in certain tasks (e.g. **100%** accuracy on the **36** pick tasks).

We run additional speed tests, comparing SARA-based ViT of the RT-2 model's PaLI backbone with its regular counterpart for images of different resolutions (on CPUs). The results are presented in Fig. 8. The ViT encoder of PaLI is computationally bottlenecked by its attention module. SARA remains a feasible approach even for high resolution images, while the regular variant does not. We plan to exercise this feature of SARA by using much higher resolution images (a challenge for regular RT-2 models) in future work.

V. CONCLUSION

We introduced SARA-RT, a new paradigm for adapting Transformer-based robotic controllers to their more practical linear-attention deployable counterparts. We have provided its comprehensive evaluation on real robots, including improving the inference of the VLA class of RT-2 models and controllers applying Point Cloud Transformers. We consider it one of the first steps towards addressing practical challenges of using Transformer-based robotic controllers.

REFERENCES

[1] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," 2023.

[2] M. Guo, J. Cai, Z. Liu, T. Mu, R. R. Martin, and S. Hu, "PCT: point cloud transformer," *Comput. Vis. Media*, vol. 7, no. 2, pp. 187–199, 2021. [Online]. Available: https://doi.org/10.1007/s41095-021-0229-5

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 5998–6008.

[4] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: Scaling language modeling with pathways," *CoRR*, vol. abs/2204.02311, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2204.02311

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[6] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus, "Emergent abilities of large language models," *Trans. Mach. Learn. Res.*, vol. 2022, 2022. [Online]. Available: https://openreview.net/forum?id=yzkSU5zdwD

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 8748–8763. [Online]. Available: http://proceedings.mlr.press/v139/radford21a.html

[8] J. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. L. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, and K. Simonyan, "Flamingo: a visual language model for few-shot learning," in *NeurIPS*, 2022.

[9] X. Chen, X. Wang, S. Changpinyo, A. J. Piergiovanni, P. Padlewski, D. Salz, S. Goodman, A. Grycner, B. Mustafa, L. Beyer, A. Kolesnikov, J. Puigcerver, N. Ding, K. Rong, H. Akbari, G. Mishra, L. Xue, A. V. Thapliyal, J. Bradbury, W. Kuo, M. Seyedhosseini, C. Jia, B. K. Ayan, C. Riquelme, A. Steiner, A. Angelova, X. Zhai, N. Houlsby, and R. Soricut, "Pali: A jointly-scaled multilingual language-image model," *CoRR*, vol. abs/2209.06794, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2209.06794

[10] A. Zeng, M. Attarian, B. Ichter, K. M. Choromanski, A. Wong, S. Welker, F. Tombari, A. Purohit, M. S. Ryoo, V. Sindhwani, J. Lee, V. Vanhoucke, and P. Florence, "Socratic models: Composing zero-shot multimodal reasoning with language," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/pdf?id=G2Q2Mh3avow

[11] B. Ichter, A. Brohan, Y. Chebotar, C. Finn, K. Hausman, A. Herzog, D. Ho, J. Ibarz, A. Irpan, E. Jang, R. Julian, D. Kalashnikov, S. Levine, Y. Lu, C. Parada, K. Rao, P. Sermanet, A. Toshev, V. Vanhoucke, F. Xia, T. Xiao, P. Xu, M. Yan, N. Brown, M. Ahn, O. Cortes, N. Sievers, C. Tan, S. Xu, D. Reyes, J. Rettinghouse, J. Quiambao, P. Pastor, L. Luu, K. Lee, Y. Kuang, S. Jesmonth, N. J. Joshi, K. Jeffrey, R. J. Ruano, J. Hsu, K. Gopalakrishnan, B. David, A. Zeng, and C. K. Fu, "Do as I can, not as I say: Grounding language in robotic affordances," in *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 2022, pp. 287–318. [Online]. Available: https://proceedings.mlr.press/v205/ichter23a.html

[12] A. Zeng, B. Ichter, F. Xia, T. Xiao, and V. Sindhwani, "Demonstrating large language models on robots," in *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, Eds., 2023. [Online]. Available: https://doi.org/10.15607/RSS.2023.XIX.024

[13] A. Z. Ren, A. Dixit, A. Bodrova, S. Singh, S. Tu, N. Brown, P. Xu, L. Takayama, F. Xia, J. Varley, Z. Xu, D. Sadigh, A. Zeng, and A. Majumdar, "Robots that ask for help: Uncertainty alignment for large language model planners," 2023.

[14] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, P. Sermanet, T. Jackson, N. Brown, L. Luu, S. Levine, K. Hausman, and B. Ichter, "Inner monologue: Embodied reasoning through planning with language models," in *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 2022, pp. 1769–1782. [Online]. Available: https://proceedings.mlr.press/v205/huang23c.html

[15] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," *CoRR*, vol. abs/2209.07753, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2209.07753

[16] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, and P. Florence, "Palm-e: An embodied multimodal language model," *ICML 2023*, vol. abs/2303.03378, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.03378

[17] J. von Oswald, E. Niklasson, E. Randazzo, J. Sacramento, A. Mordvintsev, A. Zhmoginov, and M. Vladymyrov, "Transformers learn in-context by gradient descent," *CoRR*, vol. abs/2212.07677, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2212.07677

[18] Y. Bai, F. Chen, H. Wang, C. Xiong, and S. Mei, "Transformers as statisticians: Provable in-context learning with in-context algorithm selection," *CoRR*, vol. abs/2306.04637, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.04637

[19] S. Mirchandani, F. Xia, P. Florence, B. Ichter, D. Driess, M. G. Arenas, K. Rao, D. Sadigh, and A. Zeng, "Large language models as general pattern machines," 2023.

[20] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. S. Ryoo, G. Salazar, P. R. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "RT-1: robotics transformer for real-world control at scale," in *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, Eds., 2023. [Online]. Available: https://doi.org/10.15607/RSS.2023.XIX.025

[21] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. Turner, E. Undersander, W. Galuba, A. Westbury,

A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (HM3D): 1000 large-scale 3d environments for embodied AI," *CoRR*, vol. abs/2109.08238, 2021. [Online]. Available: https://arxiv.org/abs/2109.08238

[22] K. M. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlós, P. Hawkins, J. Q. Davis, A. Mohiuddin, L. Kaiser, D. B. Belanger, L. J. Colwell, and A. Weller, "Rethinking attention with performers," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=Ua6zuk0WRH

[23] H. Peng, N. Pappas, D. Yogatama, R. Schwartz, N. A. Smith, and L. Kong, "Random feature attention," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=QtTKTdVrFBB

[24] A. Gupta, Y. Yuan, Y. Zhou, and C. Mendis, "Flurka: Fast fused low-rank & kernel attention," *CoRR*, vol. abs/2306.15799, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2306.15799

[25] L. Zheng, J. Yuan, C. Wang, and L. Kong, "Efficient attention via control variates," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. [Online]. Available: https://openreview.net/pdf?id=G-uNfHKrj46

[26] I. Schlag, K. Irie, and J. Schmidhuber, "Linear transformers are secretly fast weight programmers," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 9355–9366. [Online]. Available: http://proceedings.mlr.press/v139/schlag21a.html

[27] K. Irie, I. Schlag, R. Csordás, and J. Schmidhuber, "Going beyond linear transformers with recurrent fast weight programmers," in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021, pp. 7703–7717. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/3f9e3767ef3b10a0de4c256d7ef9805d-Abstract.html

[28] V. Likhosherstov, K. M. Choromanski, K. A. Dubey, F. Liu, T. Sarlós, and A. Weller, "Chefs' random tables: Non-trigonometric random features," in *NeurIPS*, 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/df2d62b96a4003203450cf89cd338bb7-Abstract-Conference.html

[29] V. Likhosherstov, K. Choromanski, A. Dubey, F. Liu, T. Sarlós, and A. Weller, "Favor#: Sharp attention kernel approximations via new classes of positive random features," *CoRR*, vol. abs/2302.00787, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2302.00787

[30] Z. Qin, W. Sun, H. Deng, D. Li, Y. Wei, B. Lv, J. Yan, L. Kong, and Y. Zhong, "cosformer: Rethinking softmax in attention," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [Online]. Available: https://openreview.net/forum?id=Bl8CQrx2Up4

[31] S. Boucheron, G. Lugosi, and P. Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 02 2013. [Online]. Available: https://doi.org/10.1093/acprof:oso/9780199535255.001.0001

[32] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, "Evolution Strategies as a Scalable Alternative to Reinforcement Learning," *arXiv:1703.03864*, 2017.

[33] H. Mania, A. Guy, and B. Recht, "Simple random search provides a competitive approach to reinforcement learning," *arXiv preprint arXiv:1803.07055*, 2018.

[34] S. Abeyruwan, L. Graesser, D. B. D'Ambrosio, A. Singh, A. Shankar, A. Bewley, D. Jain, K. Choromanski, and P. R. Sanketi, "i-Sim2Real: Reinforcement learning of robotic policies in tight human-robot interaction loops," *Conference on Robot Learning (CoRL)*, 2022.

[35] D. B. D'Ambrosio, N. Jaitly, V. Sindhwani, K. Oslund, P. Xu, N. Lazic, A. Shankar, T. Ding, J. Abelian, E. Coumans, G. Kouretas, T. Nguyen, J. Boyd, A. Iscen, R. Mahjourian, V. Vanhoucke, A. Bewley, Y. Kuang, M. Ahn, D. Jain, S. Kataoka, O. E. Cortes, P. Sermanet, C. Lynch, P. R. Sanketi, K. Choromanski, W. Gao, J. Kangaspunta, K. Reymann, G. Vesom, S. Q. Moore, A. Singh, S. W. Abeyruwan, and L. Graesser, "Robotic table tennis: A case study into a high speed learning system," in *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, Eds., 2023. [Online]. Available: https://doi.org/10.15607/RSS.2023.XIX.006

[36] X. Chen, J. Djolonga, P. Padlewski, B. Mustafa, S. Changpinyo, J. Wu, C. R. Ruiz, S. Goodman, X. Wang, Y. Tay, S. Shakeri, M. Dehghani, D. Salz, M. Lucic, M. Tschannen, A. Nagrani, H. Hu, M. Joshi, B. Pang, C. Montgomery, P. Pietrzyk, M. Ritter, A. J. Piergiovanni, M. Minderer, F. Pavetic, A. Waters, G. Li, I. Alabdulmohsin, L. Beyer, J. Amelot, K. Lee, A. P. Steiner, Y. Li, D. Keysers, A. Arnab, Y. Xu, K. Rong, A. Kolesnikov, M. Seyedhosseini, A. Angelova, X. Zhai, N. Houlsby, and R. Soricut, "Pali-x: On scaling up a multilingual vision and language model," *CoRR*, vol. abs/2305.18565, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.18565