Dual Inheritance Evolution Strategy for Deep Neural Network Optimization

Kent Hino, Yusuke Kimura, Yue Dong, and Takahiro Shinozaki
Department of Information and Communication Engineering
Tokyo Institute of Technology
Kanagawa, Japan
www.ts.ip.titech.ac.jp

***Abstract*—Deep neural networks (DNNs) need intensive tuning of their configurations such as network structures and learning conditions. The tuning is a type of black-box optimization problem where evolutionary algorithms are applicable. A distinctive property in evolutionary optimization of DNN configurations is that there is a double structure in the optimization; the evolutionary algorithm optimizes a chromosome representing the DNN configuration while an individual DNN with the configuration learns from training data typically by back-propagation. With an aim to obtain better-optimized DNNs by evolutionary algorithms, we propose a dual inheritance evolution strategy based on an analogy to human brain evolution where gene and culture co-evolves. The proposed method is an extension of a conventional evolution strategy by introducing an additional pass to directly propagate culture or knowledge from ancestor DNNs to descendant DNNs by integrating teacher-student learning. We apply the proposed method to the automatic tuning of an end-to-end neural network-based speech recognition system. Experimental results show that the proposed method produces a smaller model with higher recognition performance than a baseline optimization based on the Covariance Matrix Adaptation Evolution Strategy (CMA-ES).**

***Index Terms*—CMA-ES, Teacher-Student Learning, dual inheritance theory, automatic speech recognition**

I. Introduction

Deep neural network (DNN) based systems are getting more and more popular replacing conventional systems by surpassing them in performance and extending the technical horizon to ever-challenging artificial intelligence tasks. DNNs need tuning of its configuration such as network structures and learning conditions before it gives full play to its ability. While the back-propagation algorithm estimates a large number of network weights from a training data set based on gradient descent given a configuration, analytical optimization of the configuration is impossible. Therefore, human specialists are spending huge effort to optimize the configuration based on try and error. Often, the development time of a DNN system is dominated by the tuning of the configuration. Moreover, the obtained performance largely depends on the craftsmanship of the person who tunes it.

There are researches to automate the tuning process by applying black-box optimization methods such as Bayesian optimization and evolutionary algorithms [1]. While they are useful to obtain better configuration than a reasonably or randomly given initial configuration, they are inefficient in that all the learned results by an individual DNN except for its fitness score are discarded and not used in its descendants.

Considering an analogy to the organisms, the optimization of the network configuration corresponds to the evolution of the chromosome that specifies the design of the brain, and the optimization of the network weight parameters is parallel to the leaning by an individual having the brain. For human beings, there is an evolution of culture that is based on learning by individuals in the population in addition to the evolution of the chromosome. The dual inheritance theory in the evolutionary biology field [2] points out that there are interactions between the evolution of the culture and the genes. It explains that the outstanding intelligence of human beings is the result of the synergistic effect of the gene-culture interaction that accelerated evolution.

Teacher-student (TS) learning or knowledge distillation [3]–[5] is a widely used strategy to train small but high-performance DNN based on transferring knowledge from a teacher DNN to a student DNN. It is empirically known that a small network often achieves better performance than normal training by becoming a student of a large network. However, empirical tuning is required to design a student network that maintains required performance while reducing the model size from the teacher DNN.

In this paper, we propose the Dual Inheritance Evolution Strategy (DI-ES) extending conventional evolution strategy inspired by the dual inheritance theory. In addition to updating the chromosome distribution based on fitness scores of individuals at each generation, the proposed method introduces another pass of information transmission from ancestor DNNs to descendant DNNs based on teacher-student learning. While the concept of the proposed method is general to any evolutionary algorithms applied to DNN, we investigate it with Covariance Matrix Adaptation Evolution Strategy (CMA-ES). CMA-ES is one of evolution strategy algorithms which is known to be efficient and easy to use [6]. To evaluate the proposed method, we optimize an End-to-End neural network based speech recognition system included in the ESPnet speech recognition toolkit [1] that is widely used in the world. However,

[1]https://espnet.github.io/espnet/

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

the proposed method is applicable to any DNN not limited to speech recognition.

The rest of this paper is organized as follows. In Section II, we review the End-to-End Deep Neural Network based speech recognition system. In Section III, we introduce the teacher-student learning and its extensions. In Section IV, we summarize the related algorithms and describe our proposed method. In Section V, we apply the proposed method to automatically tune the End-to-End speech recognition system. We describe the experimental setup in Section VI and show the results in Section VII. Finally, we make a summary and conclude this paper in Section VIII.

II. Speech Recognition

An automatic speech recognition system takes a waveform signal of an utterance and outputs a character sequence of corresponding text as the recognition result. Typically, frequency analysis is first applied to the waveform signal with a sliding analysis window as a pre-processing, and a sequence of feature vectors is obtained [7]. Let $\boldsymbol{o} = \langle o_1, o_2, \cdots, o_T \rangle$ be a sequence of the feature vectors of length T extracted from an utterance, and $\boldsymbol{y} = \langle y_1, y_2, \cdots, y_N \rangle$ be a character sequence of length N. Then, speech recognition is formulated as a problem of finding $\hat{\boldsymbol{y}}$ that maximizes the conditional probability $P(\boldsymbol{y}|\boldsymbol{o})$ as shown in Equation (1).

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y}}{\operatorname{argmax}} P(\boldsymbol{y}|\boldsymbol{o}). \tag{1}$$

While hidden Markov model has long been used to model the probability combined with a language model by decomposing the probability using the Bayes' theorem [8]–[10], a rapidly emerging approach is to directly model the probability using an End-to-End neural network.

One approach to end-to-end speech recognition is the attention-based encoder-decoder model [11]. The framework of attention-based encoder-decoder speech recognition model consists of two RNNs: encoder and attention decoder. Encoder transforms the input acoustic frames $\boldsymbol{o}$, to a high-level hidden vector $\boldsymbol{h}$, then attention decoder decodes this hidden vector by producing the probability distribution over output y_u, conditioned on hidden vector $\boldsymbol{h}$ and previous context $y_{1:u-1}$ [12], as expressed in Equation 2.

$$\begin{aligned} P(\boldsymbol{y}|\boldsymbol{o}) &= \prod_{u=1}^{N} (y_u|\boldsymbol{h}, y_{1:u-1}), \\ y_u &\sim \text{AttentionDecoder}(\boldsymbol{h}, y_{1:u-1}), \\ \boldsymbol{h} &= \text{Encoder}(\boldsymbol{o}). \end{aligned} \tag{2}$$

Hybrid CTC-attention encoder-decoder model [13] is an extension of the attention-based encoder-decoder model by integrating Connectionist Temporal Classification (CTC) [14]. Figure 1 illustrates the structure of the hybrid CTC-attention encoder-decoder model.

In general, neural networks have many weight parameters, and they are estimated from a training set using back-propagation which is based on gradient descent. However, the relationship between the network structure such as the number of units in a hidden layer and the performance of the network is complex and analytic optimization is impossible. The learning conditions such as the termination criteria of the back-propagation also need empirical tuning.

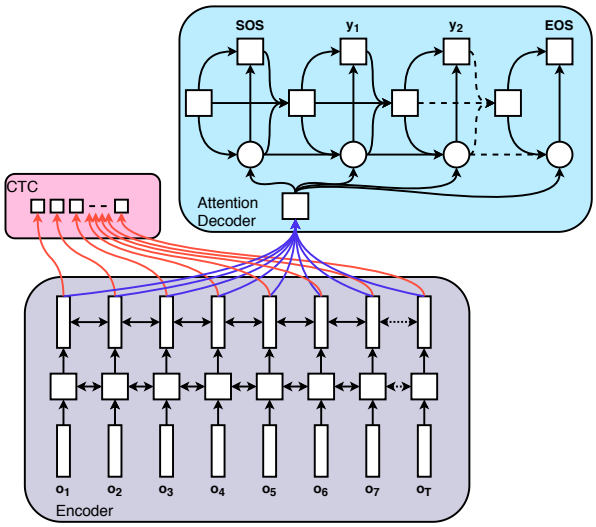

Fig. 1. Structure of End-to-End neural network speech recognition system based on Hybrid CTC-attention encoder-decoder model.

III. Teacher-Student learning

TS learning was first proposed in [15] as a method to compress DNN size by minimizing Kullback-Leibler divergence between output distributions of large-size teacher DNN and small-size student DNN. Let $P_T(c|\boldsymbol{o})$ and $P_S(c|\boldsymbol{o})$ be posterior distributions of an output category c given input $\boldsymbol{o}$ by the teacher and student DNNs, respectively. The KL divergence is defined by Equation (3).

$$\sum_c P_T(c|\boldsymbol{o}) \log \left(\frac{P_T(c|\boldsymbol{o})}{P_S(c|\boldsymbol{o})} \right). \tag{3}$$

Since the approach is to study a student model which approximates a teacher model, only the parameters of the student model are optimized. Therefore, the minimization of Equation (3) is simplified to minimization of cross-entropy shown in Equation (4).

$$-\sum_c P_T(c|\boldsymbol{o}) \log \left(P_S(c|\boldsymbol{o}) \right). \tag{4}$$

An extension to the TS learning is FitNet [16], in which additional information from Teacher's hidden layers, named *hints*, are transmitted to guide the student's learning. Since the number of neuron units in hidden layers is not always the same between the teacher and the student, a projection is introduced.

Another extension of the TS learning is the introduction of temperature T_T to the output distribution of the teacher DNN as shown in Equation (5) [3].

$$\sigma(\mathbf{c})_j = \frac{e^{c_j}/T_T}{\sum_{k=1}^{K} e^{c_k}/T_T}, \tag{5}$$

where σ is the softmax function. A higher value of T_T produces a softer probability distribution over classes. It is used with the cross-entropy loss.

IV. Baseline and Proposed Evolution Strategies

A. Covariance matrix adaptation evolution strategy

Covariance matrix adaption evolution strategy (CMA-ES) is one of the high-performance evolution strategies [6], [17]–[19]. It uses a D-dimensional real vector $\boldsymbol{x}$ as a chromosome encoding of meta-parameters to tune. By using $\boldsymbol{x}$, the tuning of the meta-parameters is formulated as a maximization problem of a black-box function $y = f(\boldsymbol{x})$, where y is the evaluation score of $\boldsymbol{x}$. Because $f(\boldsymbol{x})$ is a black-box function, we can not analytically obtain its gradient. Instead, CMA-ES defines an expected value of y based on a multivariate Gaussian distribution having a parameter set $\boldsymbol{\theta}$ as shown in Equation (6), and maximizes the expected value by optimizing $\boldsymbol{\theta}$.

$$\mathbb{E}[f(\boldsymbol{x})|\boldsymbol{\theta}], \boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}, \tag{6}$$

where $\boldsymbol{\mu}$ is a mean vector and $\boldsymbol{\Sigma}$ is a covariance matrix.

CMA-ES iteratively optimizes the expected value by using the gradient ascent method with natural gradient [20]. By using the log-trick $\nabla \log f = \frac{\nabla f}{f}$ and the sample approximation, the natural gradient is estimated by Equation (7).

$$\nabla_{\boldsymbol{\theta}} \mathbb{E}[f(\boldsymbol{x})|\boldsymbol{\theta}] \approx \frac{1}{K} \sum_{k}^{K} y_k \boldsymbol{F}_{\boldsymbol{\theta}}^{-1} \nabla_{\boldsymbol{\theta}} \log \mathcal{N}(\boldsymbol{x}_k|\boldsymbol{\theta}), \tag{7}$$

where $\boldsymbol{x}_k$ is k-th chromosome drawn from the Gaussian distribution, and $\boldsymbol{F}$ is the Fisher information matrix defined by Equation (8).

$$\boldsymbol{F}(\boldsymbol{\theta}) = \int \mathcal{N}(\boldsymbol{x}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log \mathcal{N}(\boldsymbol{x}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log \mathcal{N}(\boldsymbol{x}|\boldsymbol{\theta})^T d\boldsymbol{x}. \tag{8}$$

By substituting the Gaussian distribution $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ into Equation (7), we can obtain the update formula for $\hat{\boldsymbol{\mu}}_n$ and $\hat{\boldsymbol{\Sigma}}_n$:

$$\begin{cases} \hat{\boldsymbol{\mu}}_n = \hat{\boldsymbol{\mu}}_{n-1} + \epsilon_{\boldsymbol{\mu}} \sum_{k=1}^{K} w(y_k)(\boldsymbol{x}_k - \hat{\boldsymbol{\mu}}_{n-1}), \\ \hat{\boldsymbol{\Sigma}}_n = \hat{\boldsymbol{\Sigma}}_{n-1} + \epsilon_{\boldsymbol{\Sigma}} \sum_{k=1}^{K} w(y_k) \\ \qquad \cdot \big((\boldsymbol{x}_k - \hat{\boldsymbol{\mu}}_{n-1})(\boldsymbol{x}_k - \hat{\boldsymbol{\mu}}_{n-1})^{\intercal} - \hat{\boldsymbol{\Sigma}}_{n-1}\big), \end{cases} \tag{9}$$

where $^{\intercal}$ is the matrix transpose and n is the generation. The value y_k in Equation (7) is approximated in Equation (9) by a weight function $w(y_k)$, which is defined by Equation (10):

$$w(y_k) = \frac{\max\{0, \log(K/2+1) - \log(\mathrm{R}(y_k))\}}{\sum_{k'=1}^{K} \max\{0, \log(K/2+1) - \log(\mathrm{R}(y_{k'}))\}} - \frac{1}{K}, \tag{10}$$

where $\mathrm{R}(y_k)$ is a ranking function that returns the descending order of y_k among $y_{1:K}$ (i.e., $\mathrm{R}(y_k) = 1$ for the highest y_k, $\mathrm{R}(y_k) = K$ for the smallest y_k, and so forth). Since this equation only considers the order of y, the updates become less sensitive to the evaluation measurements [17].

CMA-ES has been applied to automatically tune DNN-HMM-based large vocabulary speech recognition systems, and its effectiveness has been demonstrated [21]. In this research, we use CMA-ES as a baseline evolution strategy.

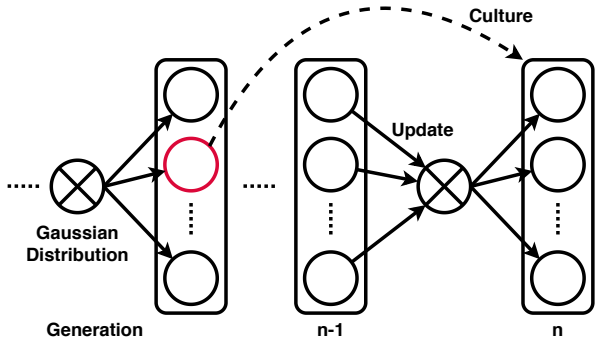

Fig. 2. Dual Inheritance Evolution Strategy.

Algorithm 1 Dual Inheritance Evolution Strategy

```
Initialize μ_0 and Σ_0
while not max_generation do
    Select a teacher DNN(T)_n from ancestors
    for k = 1 to K do
        Sample x_k from N(x|μ_{n-1}, Σ_{n-1})
        Train a student DNN(S)_k with a configuration specified by x_k and with the teacher DNN(T)_n
        Evaluate DNN(S)_k and obtain y_k
    end for
    Update μ_n, Σ_n
end while
return Best individual x and its score y = f(x)
```

B. Dual Inheritance Evolution Strategy

To propagate more information from ancestor generations to a descendant generation, we propose DI-ES that integrates TS learning into CMA-ES. Figure 2 illustrates the proposed method. At each generation n, we select a DNN from its ancestor generations and use the selected DNN as a teacher for all individuals in the current generation. Algorithm 1 summarizes the process.

In our preliminary experiment, we found there is a tendency that TS learning is not effective when a student DNN is larger than a teacher DNN. Based on the observation, we choose the teacher DNN that gives the best performance among a set of ancestor individuals whose size is larger than the initial individual.

V. DI-ES based tuning of End-to-End Speech recognition system

We apply the proposed DI-ES method to automatically tune hybrid CTC-attention encoder-decoder based speech recognition system. As the knowledge to transfer from the teacher to the student, we use encoder and decoder outputs as shown in Figure 3. Since the encoder output is a real vector while the decoder output is a categorical distribution, we use mean square loss for the encoder output based TS learning and cross-entropy loss for the decoder output based TS learning.

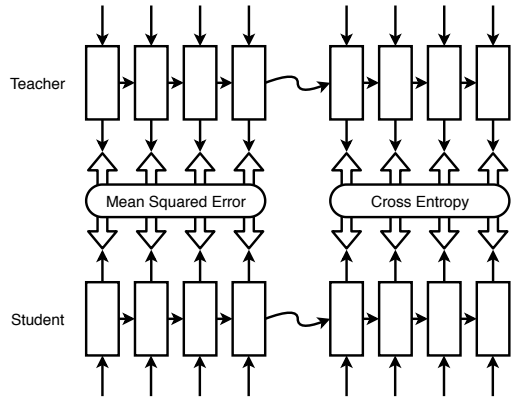

Fig. 3. TS learning of encoder-decoder model.

Equation (11) describes the total loss we use for the TS learning in DI-ES optimization.

$$\begin{aligned} Loss_{total} &= \mu \cdot Loss_{TS} + (1-\mu) \cdot Loss_{base}, \\ Loss_{TS} &= \lambda \cdot Loss_{Enc} + (1-\lambda) \cdot T_T^2 \cdot Loss_{Dec}, \\ Loss_{base} &= \alpha \cdot Loss_{ctc} + (1-\alpha) \cdot Loss_{att}, \end{aligned} \tag{11}$$

where $Loss_{base}$ is the base training loss of the hybrid CTC-attention encoder-decoder model consisting of the attention loss $Loss_{att}$ and the CTC loss $Loss_{ctc}$, and $Loss_{TS}$ is the TS learning loss consisting of the encoder TS learning loss $Loss_{Enc}$ and the decoder TS learning losses $Loss_{DEC}$. The weights μ, λ, and α are used to change the balance of the losses. Among them, μ and λ are used to control TS learning. When $\mu = 0$, the TS learning is disabled. Only decoder output based TS learning is performed when $\lambda = 0$, whereas encoder output based TS learning is performed when $\lambda = 1$. When the temperature T_T is applied to the cross-entropy loss (i.e. when T_T is not 1.0), a coefficient T_T^2 is multiplied to the decoder loss to scale the gradient [3].

In addition to minimizing the recognition error rate, reducing the DNN size is important to improve computational efficiency. If we only evaluate DNN by recognition error rate, DNN size might largely increase for a tiny reduction of the recognition error rate. In fact, we have found in our preliminary experiments that the DNN size often explodes after 5 or 6 generations if we do not consider the model size in the evolution.

To consider the model size, we use a weighted average of error rate and model size as shown in Equation (12) as the objective function of the evolution.

$$g(\boldsymbol{x}) = \gamma_1 \cdot \mathrm{Err}(\boldsymbol{x}) + \gamma_2 \cdot \frac{\mathrm{Size}(\boldsymbol{x})}{\mathrm{Size}_{init}}, \tag{12}$$

where $\mathrm{Err}(\boldsymbol{x})$ and $\mathrm{Size}(\boldsymbol{x})$ are error rate and DNN size of the ASR system built from a chromosome $\boldsymbol{x}$, and Size_{init} is DNN size of the initial individual. In our experiments, γ_1 and γ_2 are set to 1.0.

TABLE I
LIST OF META-PARAMETERS OPTIMIZED IN THE EXPERIMENT

Category	Meta-parameters	Initial value
General	patience	3
	mtlalpha	0.5
Encoder	elayers	4
	eunits	320
	eprojs	320
Decoder	dlayers	1
	dunits	300
Attentions	adim	320
	aconv-chans	10
	aconv-filts	100
TS learning	μ	0.3
	λ	0.5
	T_T	20

TABLE II
EVALUATION OF THE INITIAL MODEL

Err(train_dev)	Err(test)	Number of network weights
18.1	9.7	8,330,402

VI. EXPERIMENTAL SETUP

For speech recognition, we used ESPnet toolkit [22]. ESPnet includes many recognition systems supporting different speech databases. Among them, we used the Alphanumeric database based system (AN4). Figure 4 illustrates the details of the encoder-decoder structure of ESPnet, and structure meta-parameters we optimized by the evolution.

Table I lists all the meta-parameters including those learning conditions. Among them, structure related meta-parameters are elayers (the number of encoder layers), eunits (the number of units per an encoder layer), eprojs (projected size for the next layer), dlayers (the number of decoder layers), dunits (the number of units per a decoder layer), adim (dimension of attention vector), aconv-chans (the number of channels in attention), aconv-filts (the number of filters in attention). Learning condition related meta-parameters are patience (the number of back-propagation epochs before terminate with no loss improvement), mtlalpha (attention loss coefficient α), μ (weight of TS loss and base loss), λ (weight of encoder loss and decoder loss), T_T (temperature). Their default values are also given in the table.

For the configuration of the initial individual, we used the default setting of the AN4 recipe in the ESPnet toolkit except for α where it was set to 0.5 to enable the encoder-decoder module. Table II shows the development and test set character error rate and model size of it. We conducted four types of experiments; "**CMA-ES**" is an evolution based on conventional CMA-ES, "**DI-ES(Encoder)**" is the proposed DI-ES with encoder based TS by fixing $\lambda = 1.0$, "**DI-ES(Decoder)**" is the proposed DI-ES with decoder based TS and without temperature by fixing $\lambda = 0.0$ and $T_T = 1.0$, and "**DI-ES(Encoder+Docoder)**" is the proposed DI-ES with both encoder and decoder based TS with temperature control. In DI-ES(Encoder+Docoder), λ, and T_T are included in the

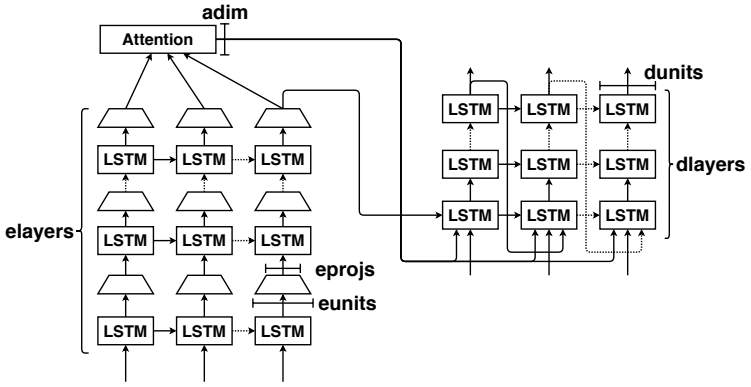

Fig. 4. ESPnet encoder-decoder network and structure meta-parameters subject for the evolutionary optimization.

chromosome, and optimized by the evolution among other meta-parameters.

The model training and evaluation were performed using TSUBAME 3.0 supercomputer [2]. Three population sizes 15, 25, and 50 were investigated. The number of generations was 15. The covariance was initialized by a unit diagonal covariance multiplied by 0.3.

VII. Results

Figure 5 shows the result by CMA-ES running 15 generations. The character error rate (CER) is defined as the number of not correctly identified characters including substitutions, insertions and deletions, divided by the total number of correct characters. The Size is defined as the the total number of DNN parameters. At generation 1, the chromosome vectors were sampled from the initial Gaussian distribution, and were distributed around the initial individual. With the progress of evolution, the distribution of the results moved to the direction of a lower error rate and a smaller number of network weights, which demonstrates the effectiveness of CMA-ES in End-to-End neural network optimization , suggesting that the automation of meta-optimization can release human specialists from manual DNN tuning processes. To the best of our knowledge, this is the first experiment that applied CMA-ES to an End-to-End speech recognition system.

Figures 6, 7, 8, 9 show results of the four strategies in the evolution experiment magnifying the region where the character error rate and the model size are smaller than the initial individual. In the figures, Pareto frontiers are also depicted. As can be seen, all the four strategies successfully reduced the character error rate and the DNN size compared to the initial individual. However, there are differences in tendencies. While the Pareto frontiers of CMA-ES with different population sizes stay almost the same position, that of the proposed DI-ES methods move toward the lower-left corner with the increase of the population size.

Figure 10 compares the Pareto frontiers of CMA-ES and proposed DI-ES methods with population size 50. It is confirmed that DI-ES methods generally produce DNNs with a smaller character error rate and a smaller number of network weights in the region. Especially, DI-ES(Encoder+Docoder) gave the best performance.

[2]https://www.gsic.titech.ac.jp/en/tsubame

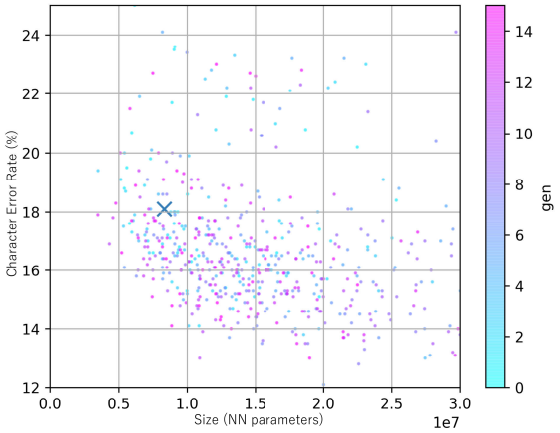

Fig. 5. Distribution of results by CMA-ES. The initial individual is marked by ×.

TABLE III
RECOGNITION RESULT

	CMA-ES	DI-ES(Enc)	DI-ES(Dec)	DI-ES(Enc+Dec)
Population	Err (train_dev set)			
15	**11.4**	12.6	12.5	13
25	12.3	12.5	**12.0**	12.5
50	12.1	**11.3**	12.2	12.6
Population	Err (test set)			
15	**5.1**	7.3	5.4	6.8
25	7.4	5.7	6.7	**4.4**
50	5.6	**4.6**	4.8	5.9

Tables III and IV summarizes the character error rates of the systems optimized by the four methods with three different population sizes. At each condition, a DNN that gives the lowest character error rate in the development set was chosen. The development and test set error rates are the result of the selected DNN. Table III is the results when the DNNs were selected without size constraint. In this case, there is no clear tendency which evolution strategy method gives the best. However, when we select DNNs from those have less than 6

Fig. 6. Results of CMA-ES

Fig. 7. Results of DI-ES: Encoder

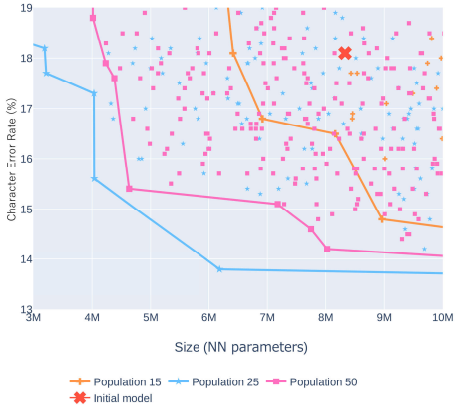

Fig. 8. Results of DI-ES: Decoder

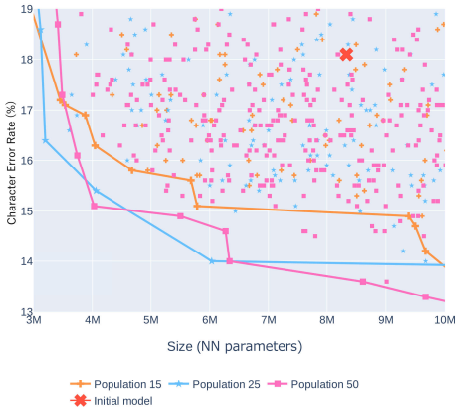

Fig. 9. Results of DI-ES: Encoder+Decoder

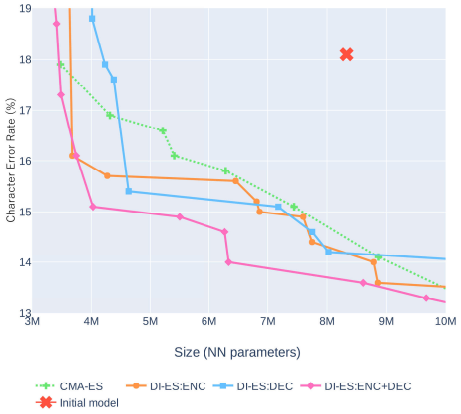

Fig. 10. Comparison of CMA-ES vs DI-ES

million weight parameters, DI-ES(Enc+Dec) gives consistently the best results for both the development and the test sets, which demonstrates the effectiveness of the proposed method to make small and high-performance DNNs.

Finally, Table V shows the tuned meta-parameters obtained by the four methods with population size 50. Compared to the initial individual obtained by default ESPnet recipe, the DNN produced by DI-ES(Enc+Dec) has higher patience, dlayers, aconv-chans. On the other hand, it has smaller eunits, eprojs, and adim, which contributed to reducing the model size.

TABLE IV
RECOGNITION RESULT(MODEL SIZE LESS THAN 6M)

	CMA-ES	DI-ES(Enc)	DI-ES(Dec)	DI-ES(Enc+Dec)
Population	Err (train_dev set) less than 6M			
15	15.9	16.7	None	**15.1**
25	15.7	15.7	15.6	**15.4**
50	16.1	15.7	15.4	**14.9**
Population	Err (test set) less than 6M			
15	9.1	9.1	None	**8.1**
25	7.4	10.0	9.8	**7.4**
50	8.4	8.1	7.9	**7.0**

VIII. Conclusion

We have proposed Dual Inheritance Evolution Strategy that integrates teacher-student learning in the framework of evolution strategy motivated by an analogy to human brain evolution, and have evaluated it using an End-to-End speech recognition system as a target of the optimization. Experimental results show that the proposed method is superior to conventional CMA-ES to produce small and high-performance DNNs. At the same time, the proposed method is the first engineering model of the dual inheritance theory which was an assumption in evolutionary biology. Future work includes

TABLE V
TUNED META-PARAMETERS

	Initial	CMA-ES	Enc	Dec	Enc+Dec
patience	3	4	5	4	7
elayers	4	3	2	3	3
eunits	320	326	362	276	263
eprojs	320	222	329	291	308
dlayers	1	2	1	1	2
dunits	300	169	69	162	280
adim	320	501	424	447	195
aconv-chans	10	10	7	11	14
aconv-filts	100	93	145	141	105
mtlalpha	1.0	0.654	0.782	0.536	0.863
μ	0.3	None	0.001	0.070	0.018
λ	0.5	None	1	0	0.058
T_T	20	None	None	1	5.678

applying our proposed method to a larger data set and transferring more general knowledge from ancestor generation to descendant generations.

ACKNOWLEDGEMENT

Part of this research was supported by The Telecommunications Advancement Foundation. Part of this work was also supported by "Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures" and "High Performance Computing Infrastructure" in Japan (Project ID: jh190066-DAH).

REFERENCES

[1] T. Moriya, T. Tanaka, T. Shinozaki, S. Watanabe, and K. Duh, "Evolution-strategy-based automation of system development for high-performance speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 1, pp. 77–88, 2019.

[2] J. Henrich and R. McElreath, "Dual-inheritance theory: the evolution of human cultural capacities and cultural evolution," in *Oxford handbook of evolutionary psychology*, 2007.

[3] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[4] J. Li, R. Zhao, Z. Chen, C. Liu, X. Xiao, G. Ye, and Y. Gong, "Developing far-field speaker system via teacher-student learning," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 5699–5703.

[5] Z. Meng, J. Li, Y. Zhao, and Y. Gong, "Conditional teacher-student learning," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6445–6449.

[6] N. Hansen, A. Auger, R. Ros, S. Finck, and P. Pošík, "Comparing results of 31 algorithms from the black-box optimization benchmarking bbob-2009," in *Proc. the 12th annual conference companion on Genetic and evolutionary computation (GECCO)*, 2010, pp. 1689–1696.

[7] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transaction on Acoustic Speech and Singal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[8] L. Bahl, P. Brown, P. de Souza, and R. Mercer, "Maximum mutual information estimation of hidden markov model parameters for speech recognition," in *ICASSP '86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 11, April 1986, pp. 49–52.

[9] M. Bisani and H. Ney, "Open vocabulary speech recognition with flat hybrid models," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[10] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4960–4964.

[11] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 577–585. [Online]. Available: http://papers.nips.cc/paper/5847-attention-based-models-for-speech-recognition.pdf

[12] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 4835–4839.

[13] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, Dec 2017.

[14] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[15] C. B. ˙a, R. Caruana, and A. Niculescu-Mizii, "Model compression," 2006. [Online]. Available: https://www.cs.cornell.edu/~caruana/compression.kdd06.pdf

[16] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6550

[17] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary Computation*, vol. 11, no. 1, pp. 1–18, 2003.

[18] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 949–980, 2014.

[19] Y. Akimoto, Y. Nagata, I. Ono, and S. Kobayashi, "Bidirectional relation between CMA evolution strategies and natural evolution strategies," in *Proc. Parallel Problem Solving from Nature (PPSN)*, 2010, pp. 154–163.

[20] S. Amari, "Natural gradient works efficiently in learning," *Neural computation*, vol. 10, no. 2, pp. 251–276, 1998.

[21] T. Moriya, T. Tanaka, T. Shinozaki, S. Watanabe, and K. Duh, "Automation of system building for state-of-the-art large vocabulary speech recognition using evolution strategy," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Dec 2015, pp. 610–616.

[22] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," in *Interspeech*, 2018, pp. 2207–2211. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1456