# Supplementary of "SCARLET-NAS: Bridging the Gap between Stability and Scalability in Weight-sharing Neural Architecture Search"

Xiangxiang Chu* Bo Zhang Qingyuan Li Ruijun Xu Xudong Li
{chuxiangxiang,zhangbo11,liqingyuan, xuruijun}@xiaomi.com, lixudong16@mails.ucas.edu.cn

## 1. Proof

*Proof.* First, we prove that Equation 3 (main text) holds for $\forall o \in \{0, 1, ..., n-2\}$. In this case, it's sufficient to prove the output of the first convolution $Conv_{(c_l,m,k,k)}$ can be exactly matched by adding $Conv_{(c_l,c_{l+1},1,1)}$ before $Conv_{(c_{l+1},m,k,k)}$. Let $W^1_{c_l,c_{l+1},1,1}$ and $W^2_{c_l,m,k,k}$ be the weight tensors of $Conv_{(c_l,c_{l+1},1,1)}$ and $Conv_{(c_{l+1},m,k,1)}$ respectively. Let $W^3_{c_l,m,k,k}$ be the weight tensors of $Conv_{(c_l,m,k,1)}$. Let $w$ be one element of the tensor. We have

$$y = Conv_{(c_l,c_{l+1},1,1)}(x_l^{c_l}), z = Conv_{(c_{l+1},m,k,1)}(y) \quad (1)$$

$$y(i,j,c) = \sum_{p=1}^{c_l} w^1_{p,c,1,1} x(i,j,p) \quad (2)$$

Also,

$$\begin{aligned} z(i,j,c) &= \sum_{q=1}^{k}\sum_{p=1}^{c_{l+1}} w^2_{p,c,q,q} y(i+q, j+q, p) \\ &= \sum_{q=1}^{k}\sum_{p=1}^{c_{l+1}} w^2_{p,c,q,q} (\sum_{u=1}^{c_l} w^1_{u,p,1,1} x(i+q, j+q, u)) \\ &= \sum_{q=1}^{k}\sum_{p=1}^{c_{l+1}}\sum_{u=1}^{c_l} w^2_{p,c,q,q} w^1_{u,p,1,1} x(i+q, j+q, u) \\ &= \sum_{q=1}^{k}\sum_{u=1}^{c_l} w^3_{u,c,q,q} x(i+q, j+q, u) \end{aligned} \quad (3)$$

Therefore, the first part is proved by setting

$$w^3_{u,c,q,q} = \sum_{p=1}^{c_{l+1}} w^2_{p,c,q,q} w^1_{u,p,1,1}. \quad (4)$$

For $o = n-1$, we replace a skip connection with an $ELS$. We can iteratively apply the first part of the proof till the end of searchable layers. □

*This work was done when all the authors were at Xiaomi AI Lab.

---

**Algorithm 1** The constrained and weighted NAS pipeline.

**Input:** Supernet $S$, the number of generations $N$, population size $n$, validation dataset $D$, constraints $C$, objective weights $w$
**Output:** A set of $K$ individuals on the Pareto front.
Train supernet $S$ defined on the scalable search space.
Uniformly generate the populations $P_0$ and $Q_0$ until each has $n$ individuals satisfying $C_{\text{FLOPS}}$, $C_{\text{Accuracy}}$.
**for** $i = 0$ **to** $N-1$ **do**
  $R_i = P_i \cup Q_i$
  $F =$ non-dominated-sorting($R_i$)
  Pick $n$ individuals to form $P_{i+1}$ by ranks and the crowding distance **weighted** by $w$.
  $Q_{i+1} = \emptyset$
  **while** $size(Q_{i+1}) < n$ **do**
    $M =$ tournament-selection($P_{i+1}$)
    $q_{i+1} =$ crossover($M$) $\cup$ hierarchical-mutation($M$)
    {Check the FLOPS constraint at first (It takes $< 1ms$).}
    **if** $FLOPS(q_{i+1}) > FLOPS_{max}$ **then**
      **continue**
    **end if**
    Evaluate model $q_{i+1}$ with $S$ on $D$ {Check the accuracy constraint (It takes $\approx 60s$).}
    **if** $Accuracy(q_{i+1}) > Acc_{min}$ **then**
      Add $q_{i+1}$ to $Q_{i+1}$
    **end if**
  **end while**
**end for**
Select $K$ equispaced models near Pareto-front from $P_N$

## 2. Algorithm

Our constrained and weighted NAS pipeline is listed in Algorithm 1 and Fig. 1.

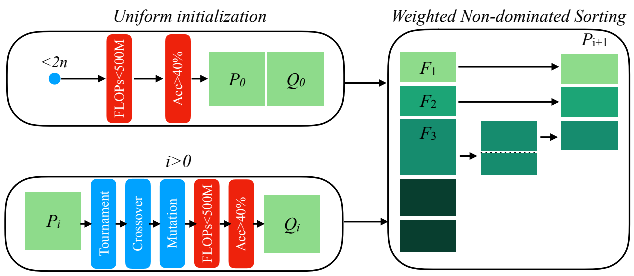



Figure 1. Constrained and weighted NSGA-II Pipeline. It starts with a uniform initialization (top left) with some constraints (red) to generate the initial population. The trained scalable supernet serves as a fast evaluator to decide the relative performance of each model so that they can be grouped into several Fronts ($F_1, F_2, \ldots$) by weighted non-dominated sorting (right). Only the top $n$ of them make up the next generation $P_{i+1}$, based on which $Q_{i+1}$ is produced with tournament selection, crossover and mutation (blue) under the same constraints (bottom left). The whole evolution loops until we reach Pareto-optimality.

# 3. Experiments

## 3.1. Search Space

For later experiments, we add skip connections to commonly used search space to construct $S_1$ and $S_2$. They are described as follows,

**Search Space $S_1$.** It is similar to ProxylessNAS [2], where MobileNetV2 [7] is adopted as its backbone. In particular, $S_1$ is represented as a block-level supernet with $L = 19$ layers of $N = 7$ choices each. Its total size is $7^{19}$. The choices are,

- MobileNetV2's inverted bottleneck blocks [7] of two expansion rates ($x$) in (3,6), three kernel sizes ($y$) in (3,5,7), labelled as MBE$x$K$y$[1],
- skip connection (the 6th choice[2]).

**Search Space $S_2$.** On top of $S_1$, we give each inverted bottleneck a squeeze-and-excitation [5] option (e.g., E$x$K$y$, E$x$K$y$_SE), similar to MnasNet [8]. Its total size thus becomes $13^{19}$.

We have to notice that skip connections are commonly used [8, 6, 1], but meticulously neglected in recent single-path one-shot methods [4, 3].

## 3.2. NSGA-II Hyperparameters

The hyperparameters for the weighted NSGA-II approach are given in Table 1.

[1] The order of numbering $o = (x - 3) + (y - 3)/2$.

[2] zero-based numbering

| Item | value | Item | value |
|---|---|---|---|
| Population N | 70 | Mutation Ratio | 0.8 |
| $p_{rm}$ | 0.2 | $p_{re}$ | 0.65 |
| $p_{pr}$ | 0.15 | $p_M$ | 0.7 |
| $p_{K-M}$ | 0.3 | | |

Table 1. Hyperparameters for the weighted NSGA-II approach.

## 3.3. More Details about Scalable Supernet with ELS

Given an input of a chickadee[3] image from ImageNet, we illustrate both high-level and low-level feature maps of the trained supernet with our proposed improvements in Figure 2. Pure skip connection easily interferes with the training process as it causes incongruence with other choice blocks. Note the channel size of feature map after Choice 6 in Figure 2 (a) is half of others because the previous channel size is 16, while other choice blocks output 32 channels. This effect is largely attenuated by ELS. As it goes deeper, we still observe consistent high-level features. Specifically, when ELS is not enforced, high-level features of deeper channels easily get blurred out, while the supernet with ELS enabled continues to learn useful features in deeper channels.

## 3.4. Search Space Evaluation

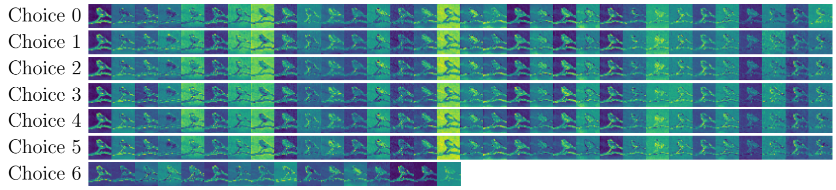
NAS results can benefit from good search space. To prove the validity of the proposed method, we show our search space has a wide range and is not particularly designed. We pick two extreme cases, one with all identity blocks (only the stem and the tail remains), the other with all K7E6s. They have the minimum and the maximum FLOPS respectively. We list their evaluation result in Table 2. The former has $24.1\%$ top-1 accuracy on ImageNet, and the latter $76.8\%$ at a cost of 557M FLOPs. Both are infeasible solutions as they violate either $acc_{min}$ or $madds_{max}$. It's thus a challenging task to deal with such search space for ordinary search techniques.

## 3.5. Analysis of SCARLET Models

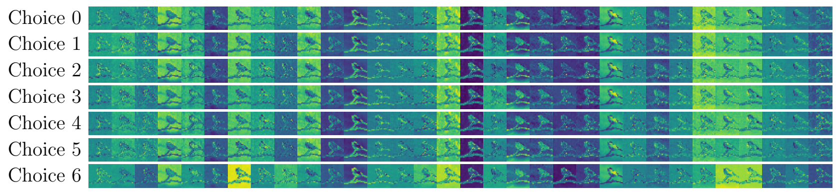
SCARLET-A makes full use of large kernels (five $5 \times 5$ and seven $7{\times}7$ kernels) to enlarge receptive field. Besides it activates many squeezing and excitation (12 out of 19) blocks to improve its classification performance. At the early stage, it appreciates either large kernels and small expansion ratios or small kernels and large expansion ratios to balance the trade-off between accuracy and FLOPs.

SCARLET-B chooses two identity operations. Compared with A, it shortens network depth at the last stages. Besides, it utilizes squeezing and excitation block extensively (14 out of 17). It places a large expansion block with large kernels at the tail stage.

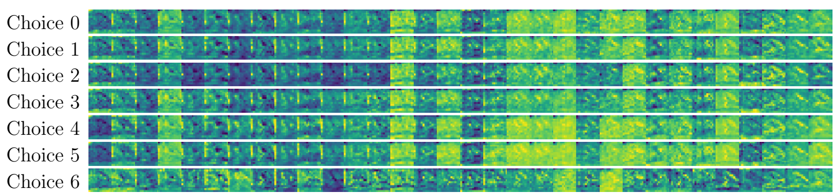
[3] ImageNet ID: n01592084_7680



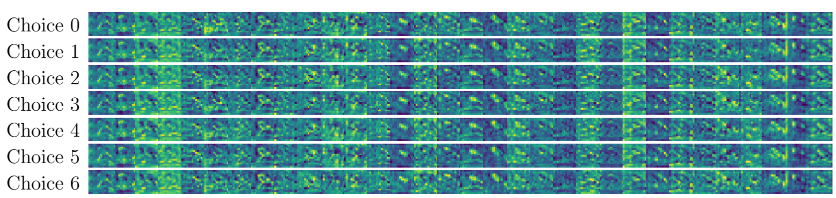

(a) First choice blocks' feature maps without ELS




(b) First choice blocks' feature maps with ELS




(c) High-level choice blocks' feature maps without ELS




(d) High-level choice blocks' feature maps with ELS

Figure 2. Learned low-level and high-level features for the supernet with and without ELS.

| Models | FLOPS (M) | $> madds_{max}$ | Top-1 (%) | Top-5 (%) | $< acc_{min}$ |
|---|---|---|---|---|---|
| All Identity | 23 | No | 24.1 | 45.0 | Yes |
| All K7E6 | 557 | Yes | 76.8 | 93.3 | No |

Table 2. Full train results of models with minimal and maximal FLOPS.

SCARLET-C uses three identity operations and utilizes small expansion ratio extensively to cut down the FLOPs, large expansion ratio at the tail stage whose resolution is $7 \times 7$. It prefers large kernels before the downsampling layers. Besides, it makes an extensive use of squeeze and excitation to boost accuracy.

## References

[1] Gabriel Bender, Pieter-Jan Kindermans, Barret Zoph, Vijay Vasudevan, and Quoc Le. Understanding and Simplifying One-Shot Architecture Search. In *ICML*, pages 549–558, 2018.

[2] Han Cai, Ligeng Zhu, and Song Han. ProxylessNAS: Direct Neural Architecture Search on Target Task and Hardware. In *ICLR*, 2019.

[3] Xiangxiang Chu, Bo Zhang, Ruijun Xu, and Jixiang Li. FairNAS: Rethinking Evaluation Fairness of Weight Sharing Neural Architecture Search. *arXiv preprint. arXiv:1907.01845*, 2019.

[4] Zichao Guo, Xiangyu Zhang, Haoyuan Mu, Wen Heng, Zechun Liu, Yichen Wei, and Jian Sun. Single Path One-Shot Neural Architecture Search with Uniform Sampling. *arXiv preprint. arXiv:1904.00420*, 2019.

[5] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-Excitation Networks. In *CVPR*, pages 7132–7141, 2018.

[6] Hanxiao Liu, Karen Simonyan, and Yiming Yang. DARTS: Differentiable Architecture Search. In *ICLR*, 2019.

[7] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *CVPR*, pages 4510–4520, 2018.

[8] Mingxing Tan, Bo Chen, Ruoming Pang, Vijay Vasudevan, and Quoc V Le. Mnasnet: Platform-Aware Neural Architecture Search for Mobile. In *CVPR*, 2019.