Multi-objective Hyperparamater Optimization of Deep Neural Networks

Kevin Duh

Johns Hopkins University

Joint work with:

Tomohiro Tanaka, Takafumi Moriya, Hao Qin, Takahiro Shinozaki (TokyoTech)

Xuan Zhang and Shinji Watanabe (JHU); Michael Denkowski (Amazon)

Success stories in Deep Learning

Computing

Facebook Creates Software That Matches Faces Almost as Well as You Do

Facebook's new AI research group reports a major improvement in face-processing software.

by Tom Simonite March 17, 2014

Advances in the relatively new artificial-intelligence field known as deep learning could fundamentally reshape what computers can do.

Asked whether two unfamiliar photos of faces show the same person, a human being will get it right 97.53 percent of the time. New software

p Learning

Computing

Facebook Creates Software That Matc

You l

Faceboo

improven

by Tom Si

Advances in
intelligence
could funda
computers c

Asked wh
human be

NATURE | NEWS

عربي

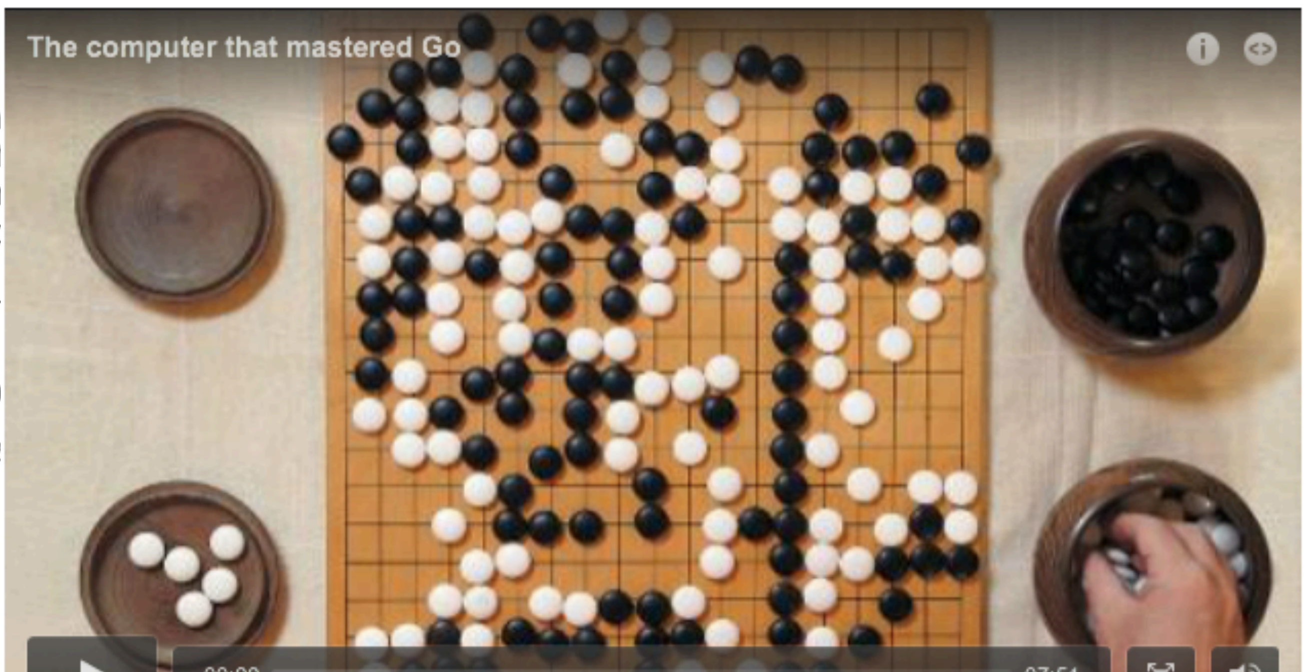
Google AI algorithm masters ancient game of Go

Deep-learning software defeats human professional for first time.

Elizabeth Gibney

27 January 2016

PDF | Rights & Permissions



rning

Computing

Facebook Creates Software That Matc
You I

Faceboo
improven

by Tom Si

Advances in
intelligence
could funda
computers

Asked wh
human be

NATURE | NEWS

عربي

Google AI algorithm masters ancient game of Go

Deep-learning software def

Elizabeth Gibney

27 January 2016

PDF | Rights & Pe

The computer that master

Packt>

Search...

News

Microsoft's new neural text-to-speech service lets machines speak like people

September 28, 2018 - 8:02 am

Microsoft has come out with a production system that performs text-to-speech (TTS) synthesis using deep neural networks. This new production system makes it hard for you to distinguish the voice of computers from human voice recordings.

The Neural text-to-speech synthesis has significantly reduced the 'listening fatigue' when talking about interaction with AI systems. It enables the system with human-like, natural sounding voice, that makes the interaction with chatbots and virtual assistants more engaging. This neural-network powered text-to-speech system was demonstrated by the Microsoft team at the Microsoft Ignite conference in Orlando, Florida, this week.

Behind each success, there are numerous unsung heroes

Massive amounts of data & compute

Countless days of trial-and-error for hyperparameter tuning

Motivation

We want an optimizer that:

1. Automates hyperparameter tuning process

1. Discovers hyperparameters that are good along multiple objectives, e.g. accurate & fast

Outline

1. Motivation
2. Problem Definition
3. Multi-objective evolutionary strategy
4. Experiment on speech recognition
5. Ongoing work

Problem Definition:
Black-box Optimization

Hyperparameter setting encoded as vector in R^d

e.g. Accuracy on Dev set

$$\begin{pmatrix} 3 \\ 200 \\ 1 \\ 0.2 \end{pmatrix} \begin{array}{l} \rightarrow \text{\# layers} \\ \rightarrow \text{\# units/layer} \\ \rightarrow \text{SGD (vs. AdaGrad)} \\ \rightarrow \text{learning rate} \end{array}$$

Problem Definition: Black-box Optimization

$$x \longrightarrow \boxed{\begin{array}{l}\text{Train Model(x)} \\ \text{on data, and} \\ \text{run on Dev set}\end{array}} \longrightarrow f(x)$$

Hyperparameter setting encoded as vector in R^d

e.g. Accuracy on Dev set

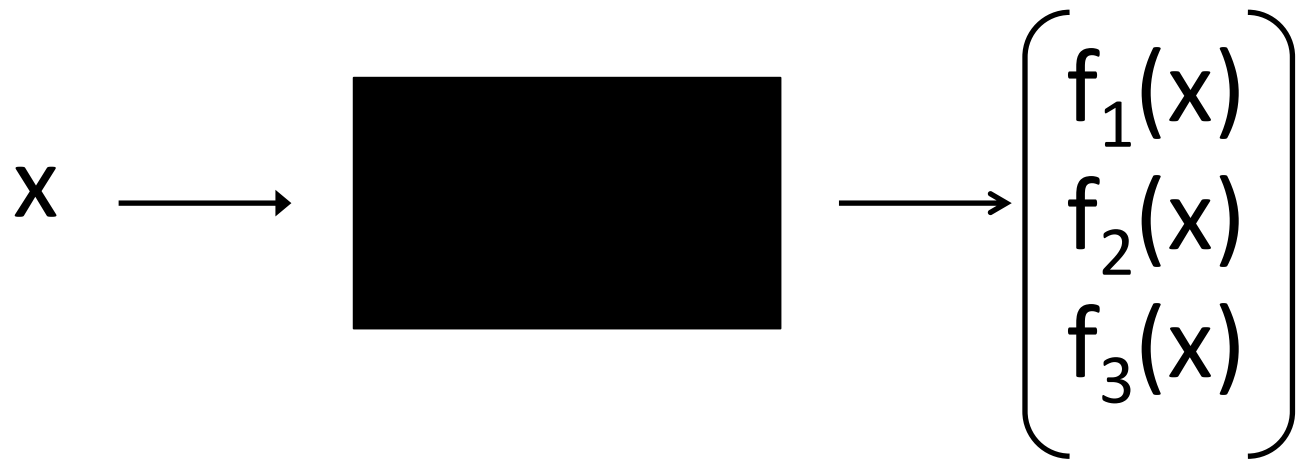
$$\begin{pmatrix} 3 \\ 200 \\ 1 \\ 0.2 \end{pmatrix} \begin{array}{l} \rightarrow \text{\# layers} \\ \rightarrow \text{\# units/layer} \\ \rightarrow \text{SGD (vs. AdaGrad)} \\ \rightarrow \text{learning rate} \end{array}$$

Problem Definition:
Black-box Optimization

Goal:

Find $x^{*}=\text{argmax}_{x} f(x)$ with few function evaluations

Problem Definition:
Black-box Optimization



Multi-objective extension, $f_i(x)$ is:

- Accuracy on Dev set (%)
- Speed of inference on Dev set (ms)
- Model size on disk (MB)

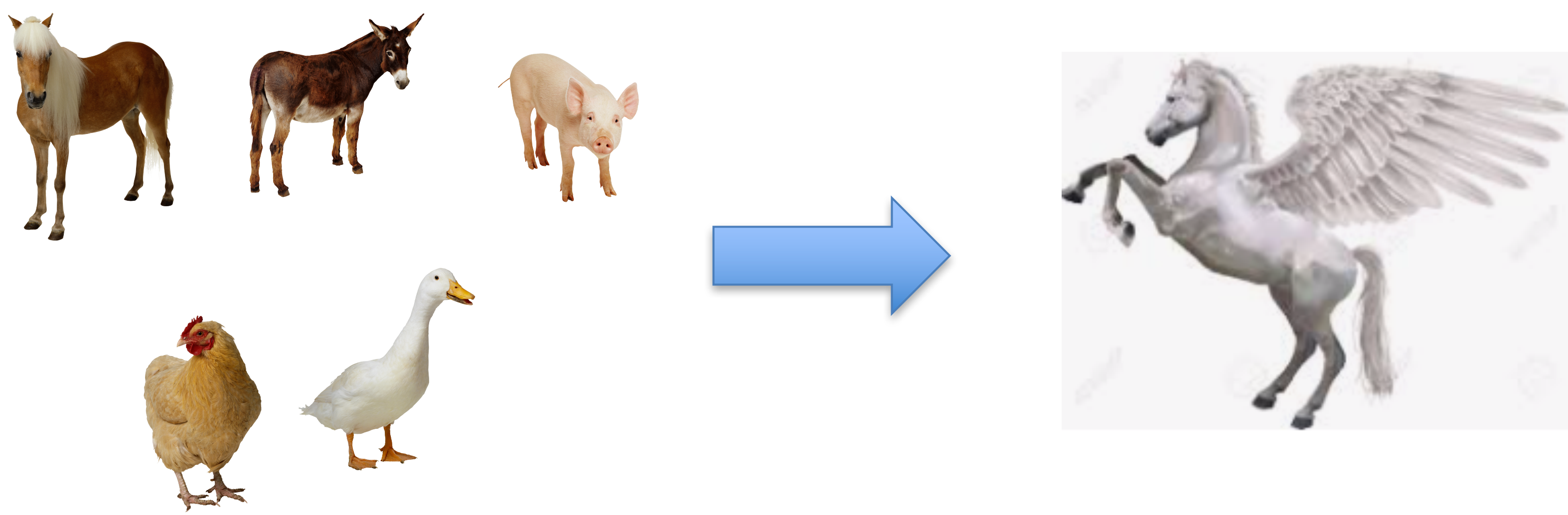
Outline

1. Motivation
2. Problem Definition
3. Multi-objective evolution strategy
4. Experiment on speech recognition
5. Related/future work

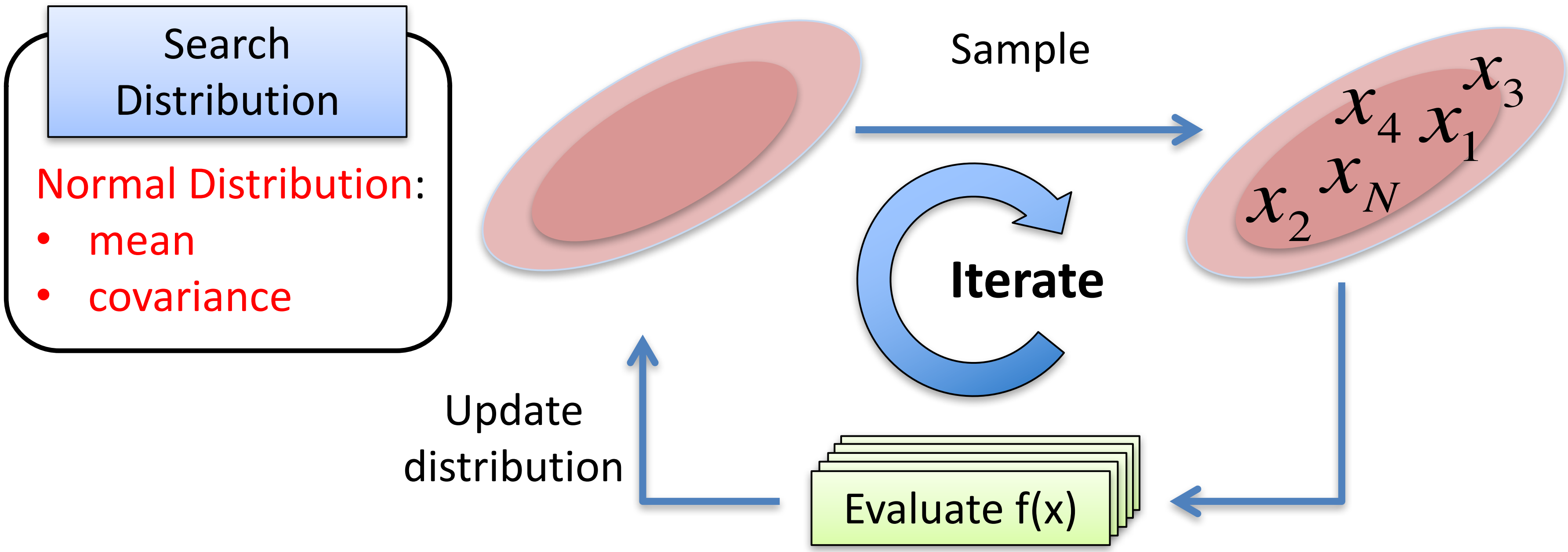
Evolutionary Strategy

1. Estimate a search distribution **P**(x) that is concentrated on regions with high fitness f(x)
2. Sample new x's based on search distribution **P**

$$x_{new} \sim P_\theta(x)$$

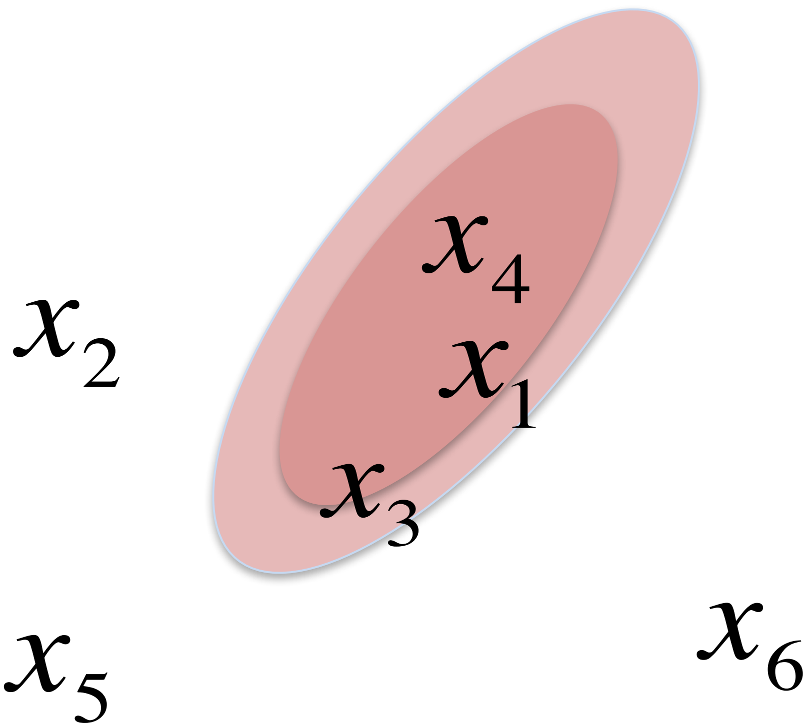


Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES)



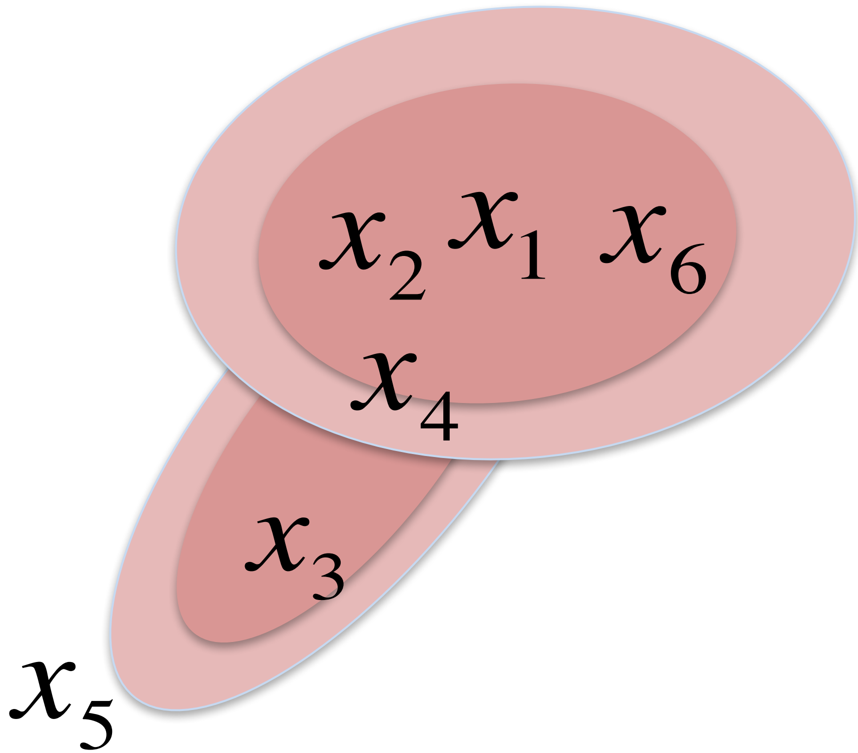
N. Hansen, S. D. Muller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," Evolutionary Computation, vol. 11, no. 1, pp. 1–18, 2003.

Intuition



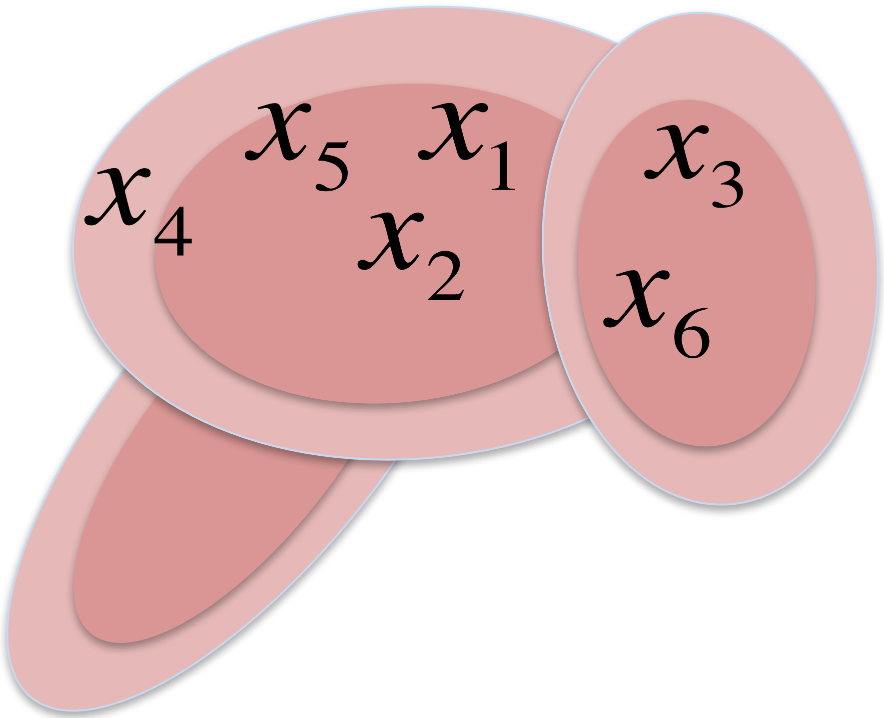
Generation 0

Intuition



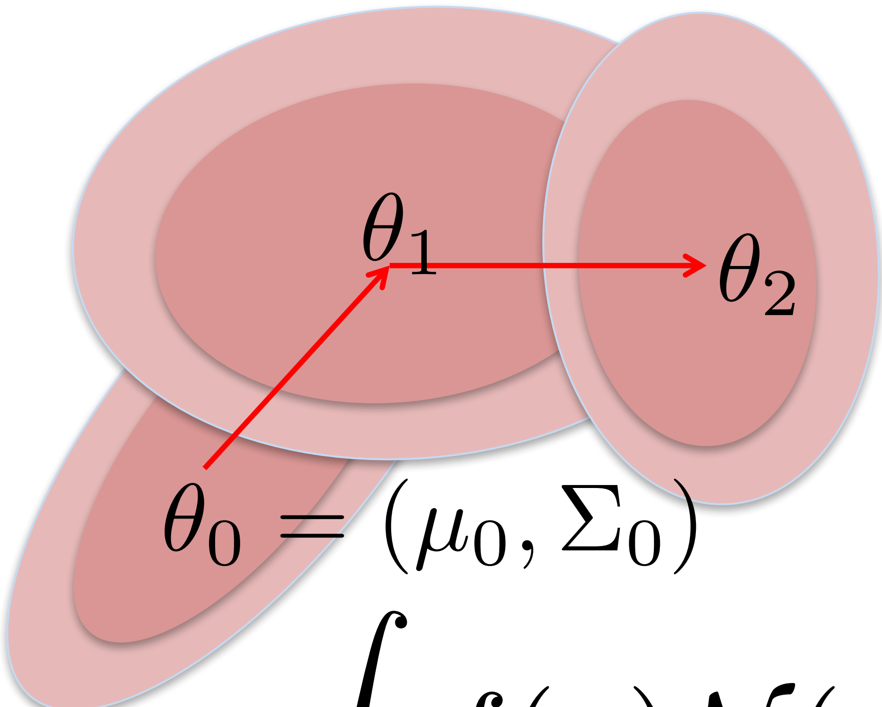
Generation 1

Intuition



Generation 2

Intuition



$$\hat{\theta} = \arg\max_{\theta} \underbrace{\int f(x)\mathcal{N}(x|\theta)dx}_{\triangleq \mathbb{E}[f(x)|\theta]}$$

Updating the search distribution

Mean:

Population size

Difference from mean to x_k

$$\hat{\mu}_n = \hat{\mu}_{n-1} + \epsilon_\mu \sum_{k=1}^{K} w(y_k)(x_k - \hat{\mu}_{n-1})$$

Mean at previous generation

Fitness of x_k, i.e. $y_k=f(x_k)$

Weight function:
More fit → higher weight

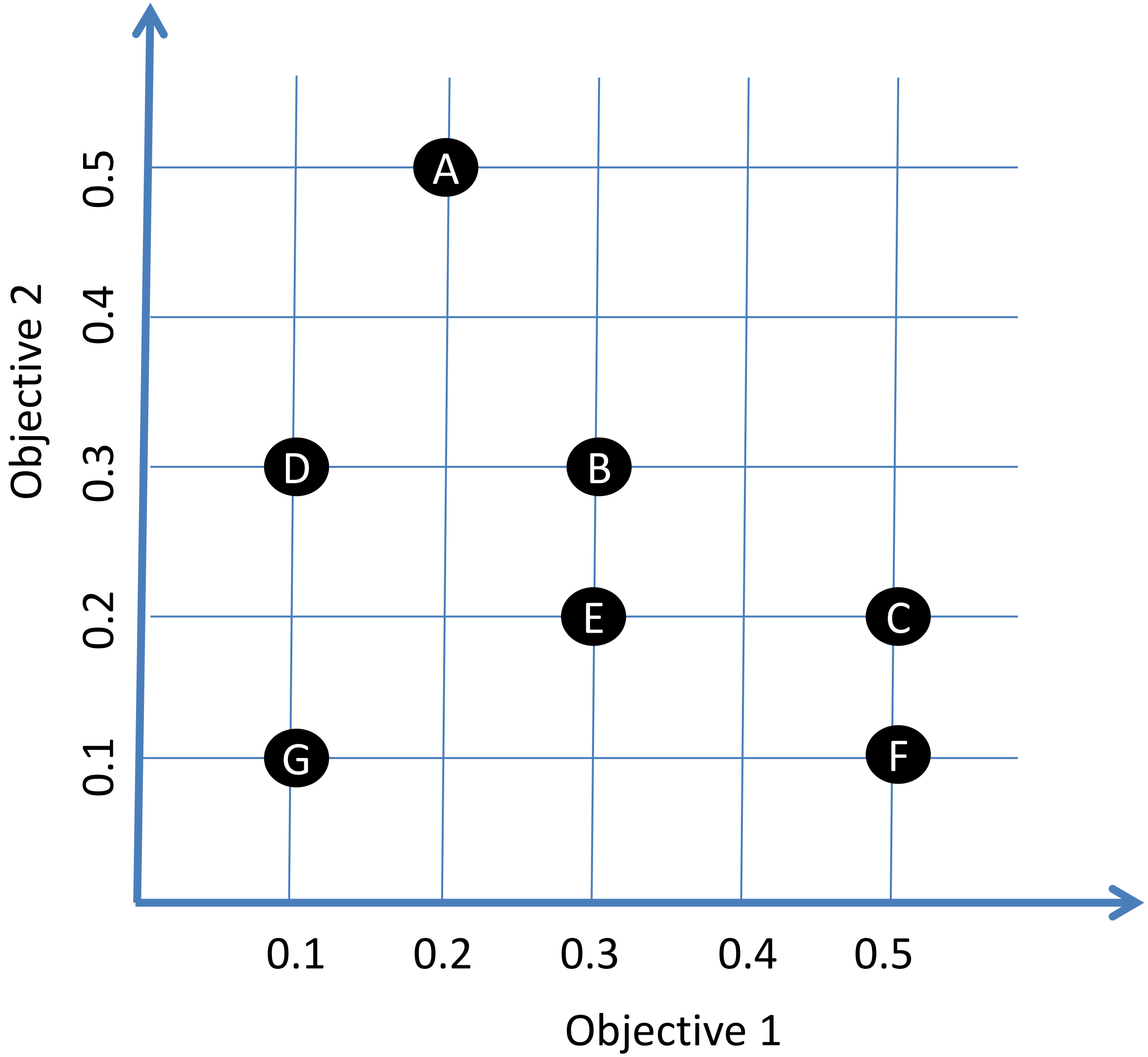
Similarly for Covariance

Multi-objective extension

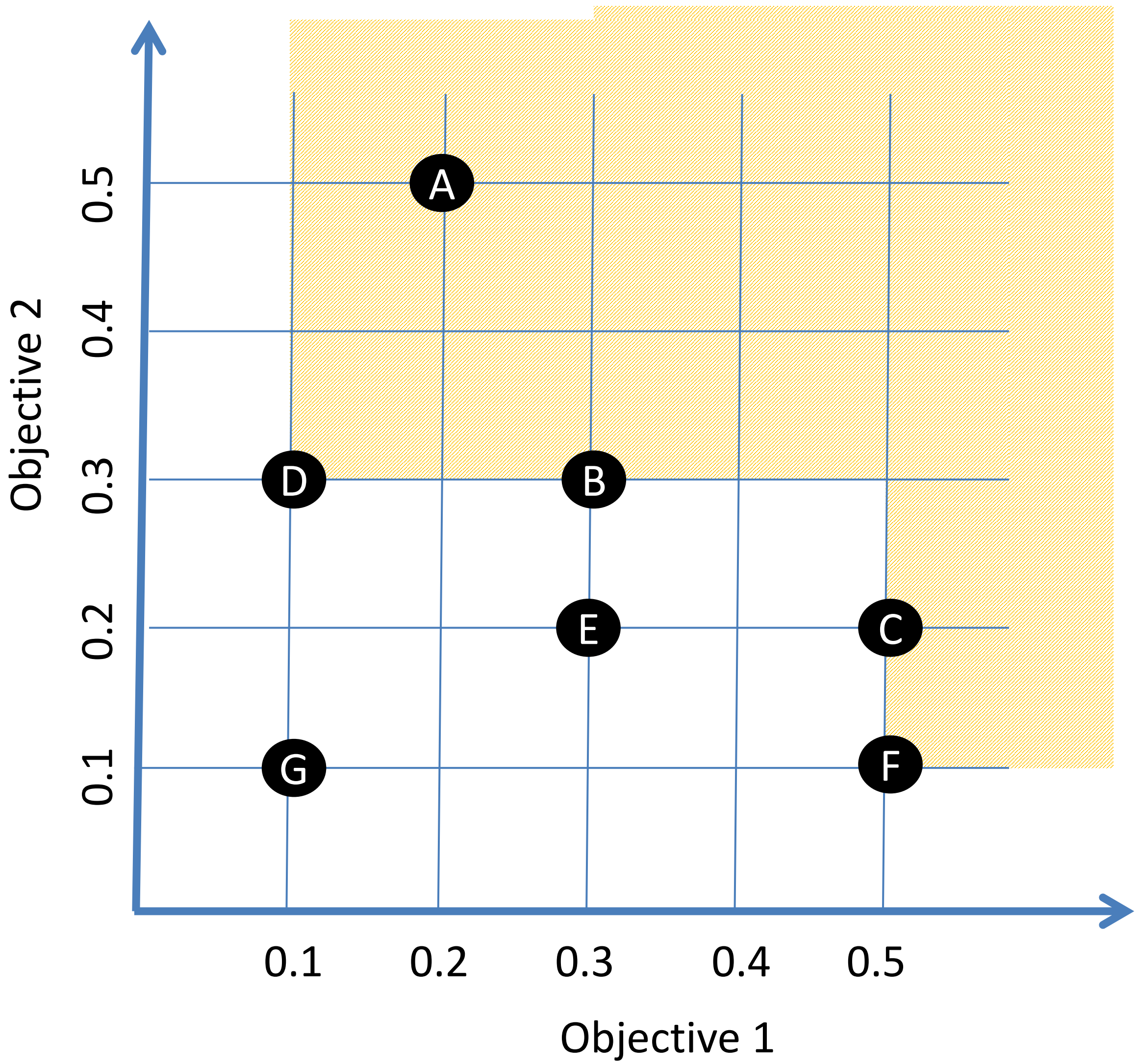
A ranking of individuals is sufficient to determine weight $w(y_k)$

How to rank under multiple objectives?

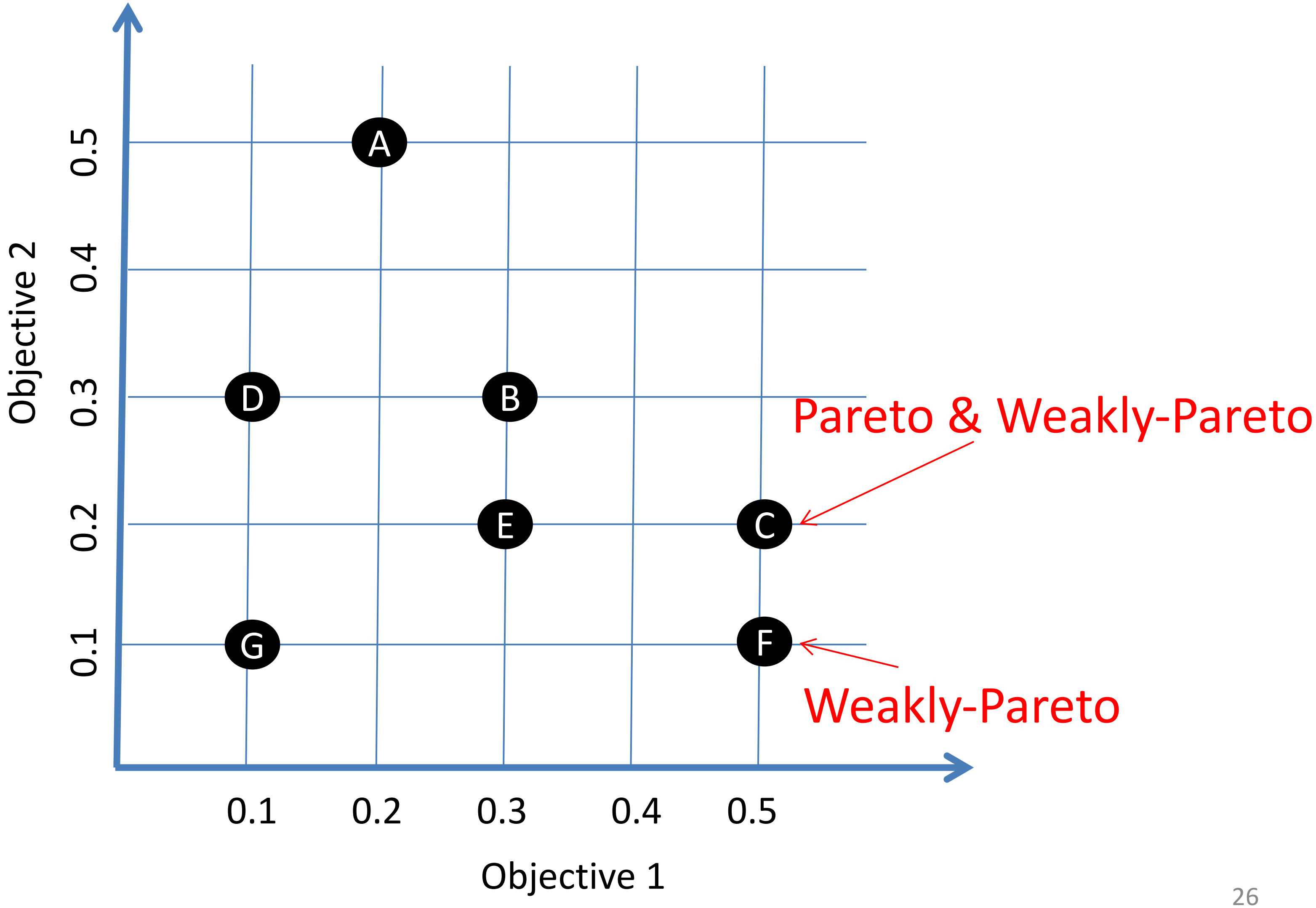
How to define optimality



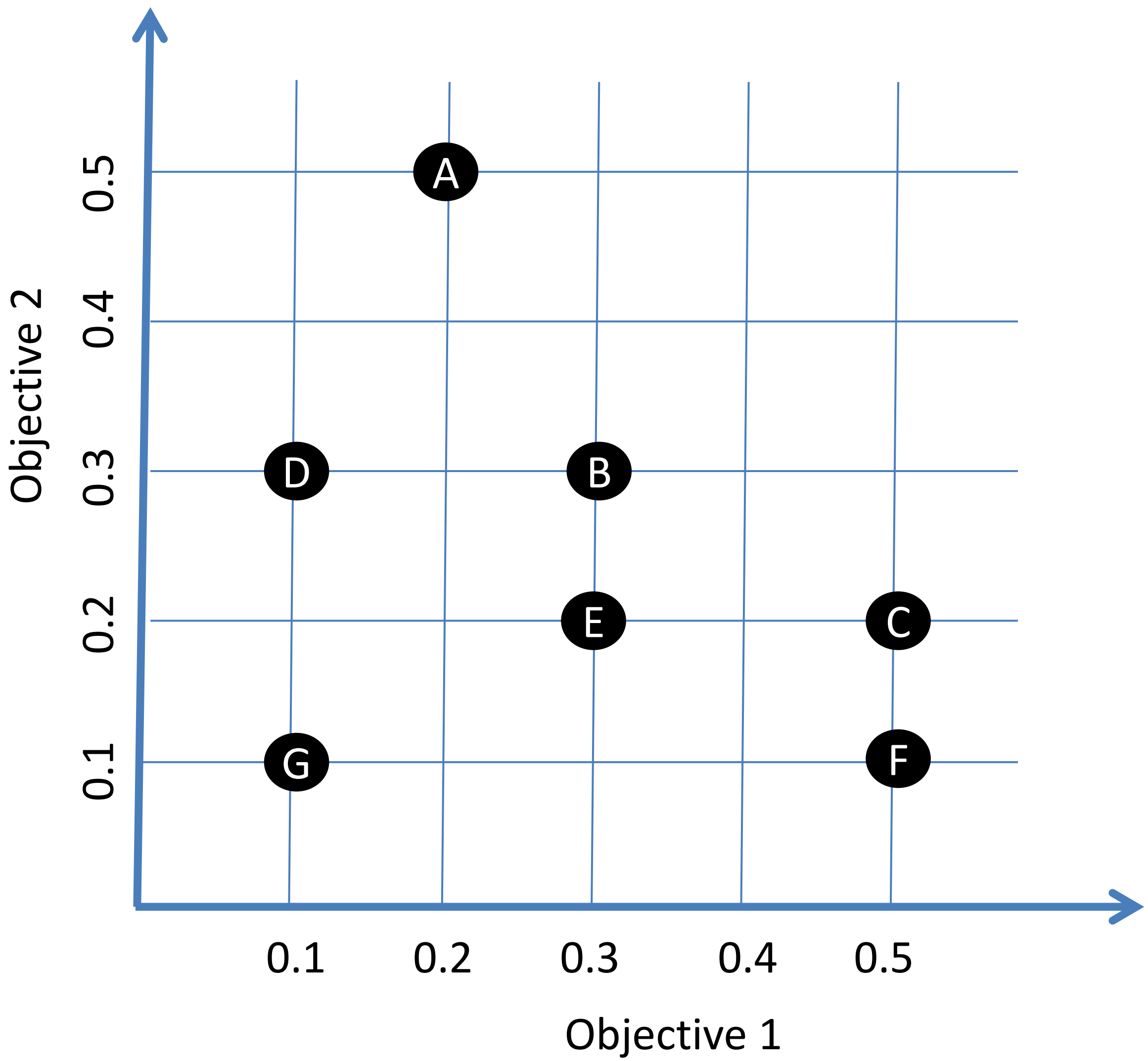
A point p is **weakly pareto-optimal** iff there does not exist another point q such that $F_k(q) > F_k(p)$ for all k



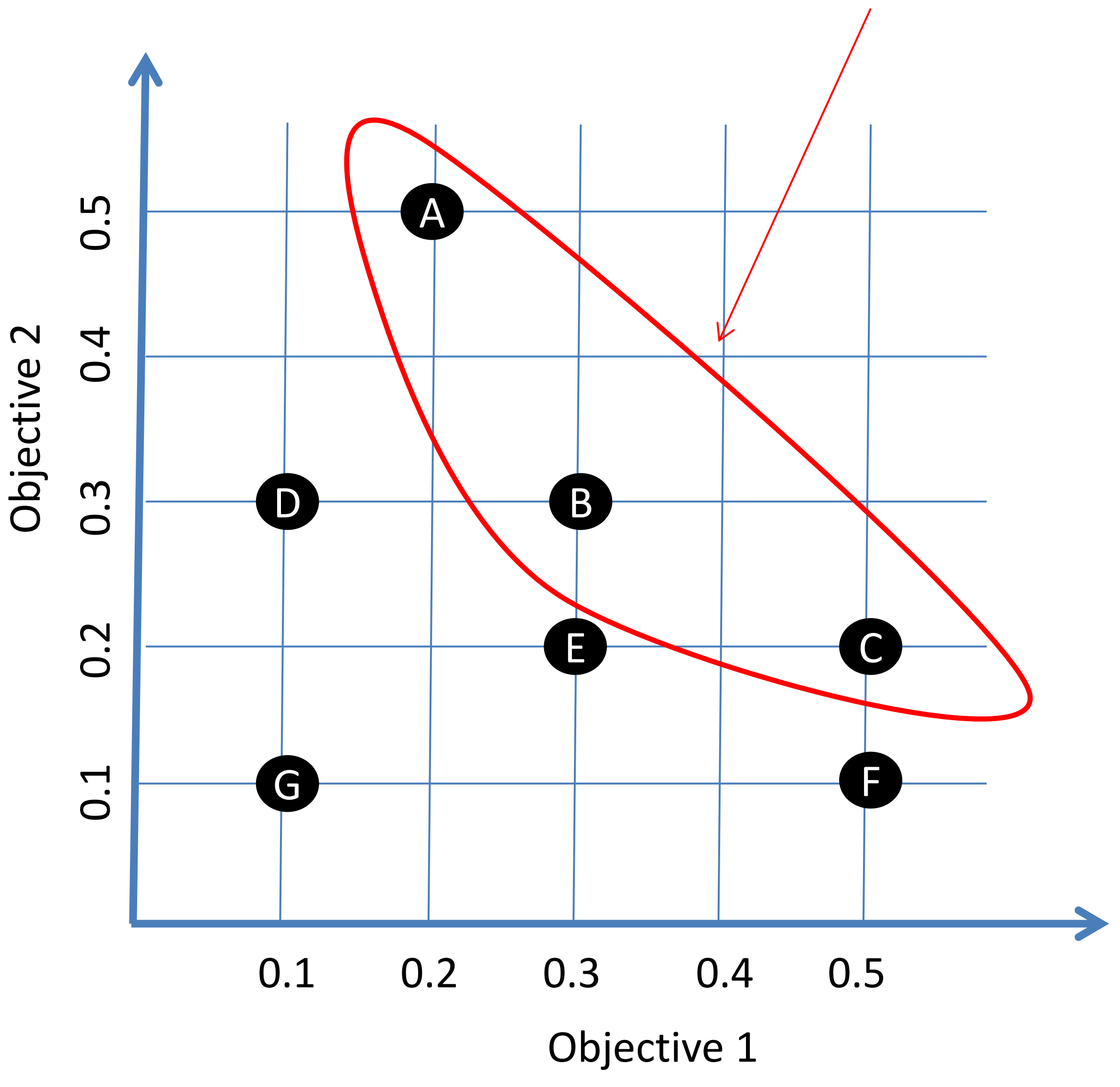
A point p is **pareto-optimal** iff there does not exist a q such that $F_k(q) >= F_k(p)$ for all k and $F_k(q) > F_k(p)$ for at least one k



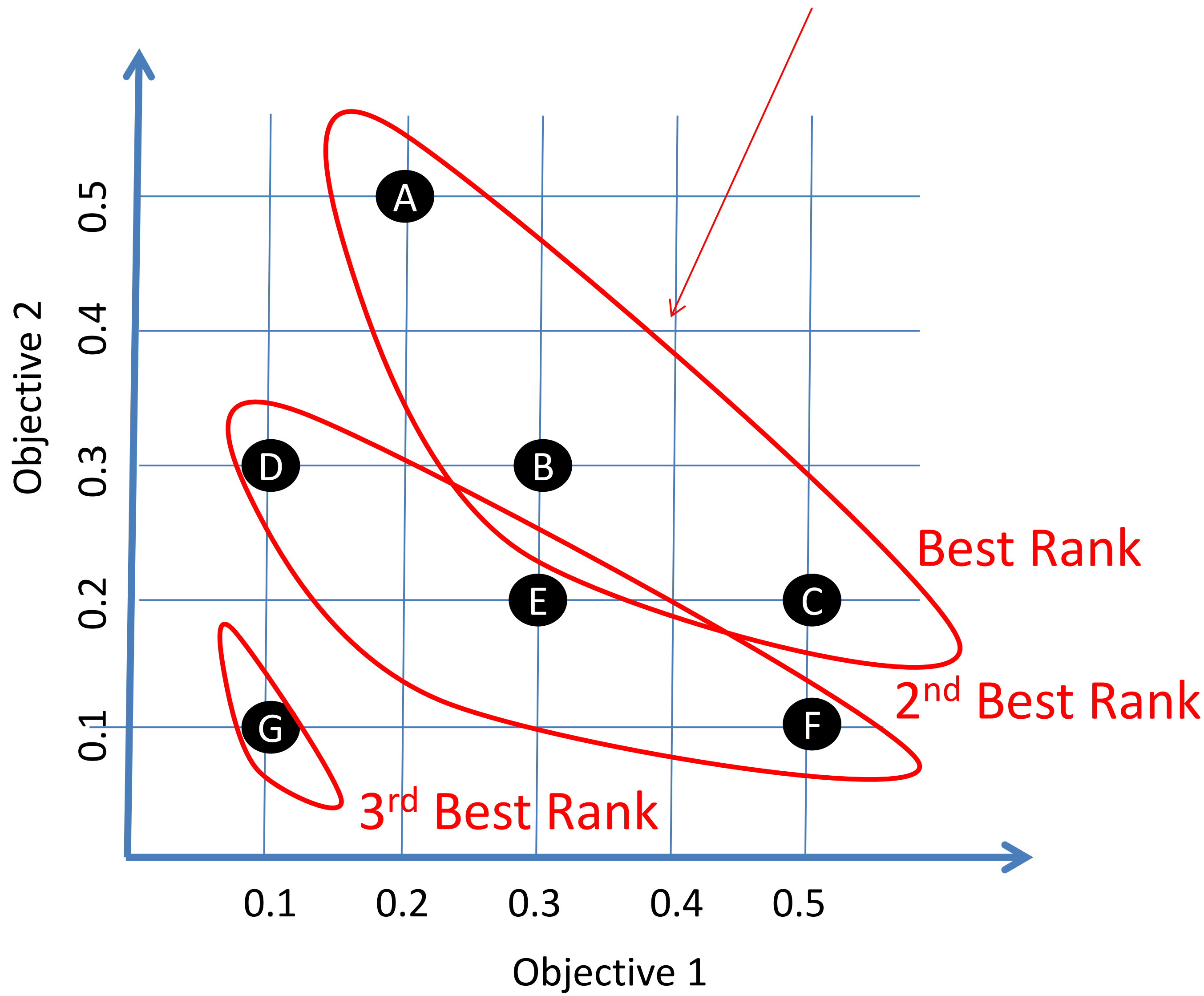
Exercise



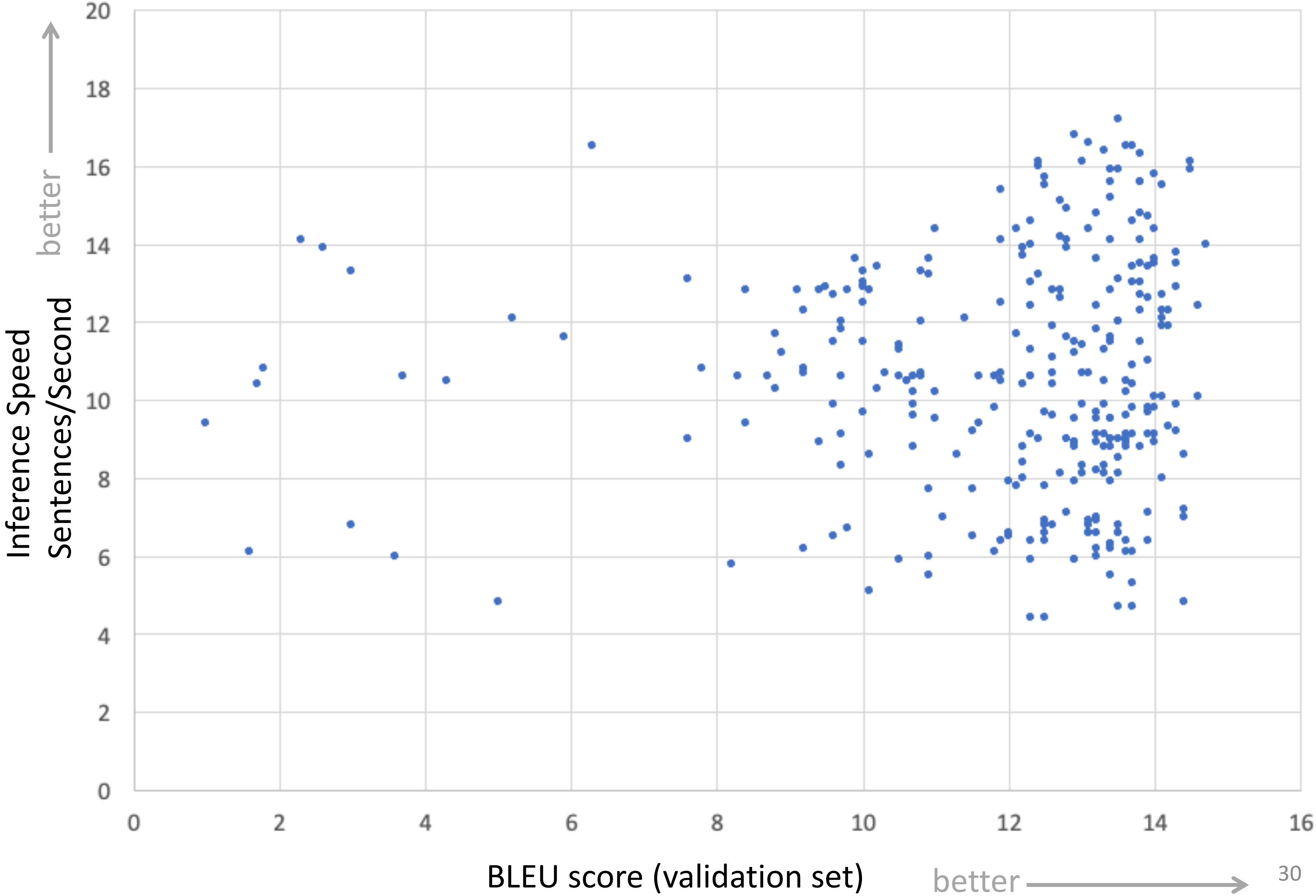
Given a set of points, the subset of pareto-optimal points form the **Pareto Frontier**



Points can be ranked by successively peeling off the **Pareto Frontier** and recomputing



Example Plot of 300 Neural Machine Translation Models with different hyperparameters (TED Zh-En)

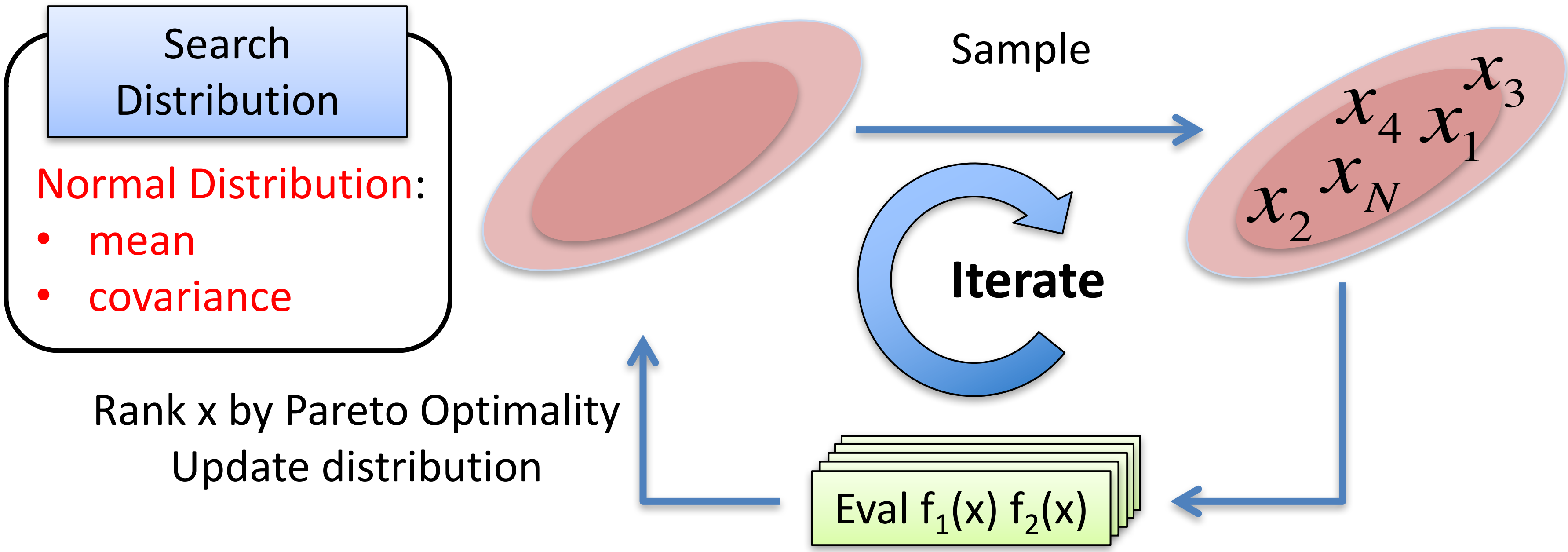


Example with 4 objectives

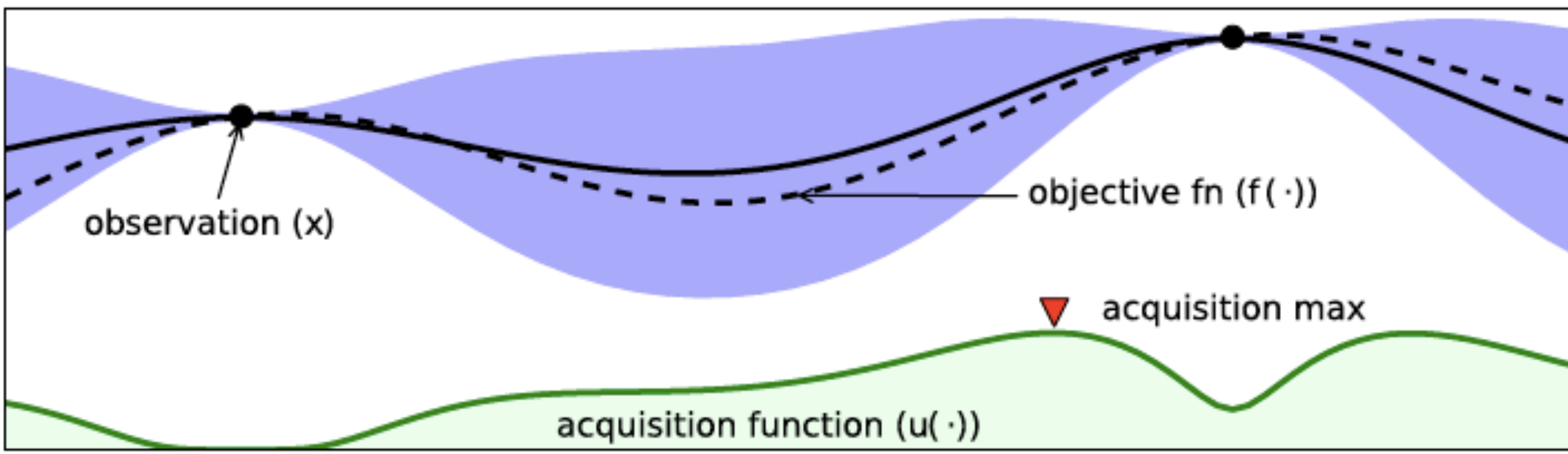
BLEU	GPU Infer Speed: sent/sec	CPU Infer Speed: sent/sec	Model Size (MB)	Hyperparameters
20.3	16.8	0.7	158	(a) RNN-LSTM, 10k BPE, 1 layer, 512 embedding
20.2	8.6	0.8	77	(b) Transformer, 10k BPE, 4 layer, 8 head, 256 embed
20.2	14.9	1.1	291	(c) RNN-LSTM, 10k BPE, 2 layer, 1024 embedding
20.2	14.0	1.6	104	(d) RNN-LSTM, 10k BPE, 2 layer, 512 embedding
20.1	7.8	0.9	77	(e) Transformer like (b), different optimizer
19.7	19.3	2.4	85	(f) RNN like (a), different optimizer
17.3	15.9	3.3	79	(g) RNN-GRU, 10k BPE, 1 layer, 512 embedding
19.4	8.1	1.5	46	(h) Transformer, 10k BPE, 2 layer, 8 head, 256 embed

Example Table of Neural Machine Translation Models with different hyperparameters (TED Ru-En) – All Pareto-optimal

Quick Summary:
Multi-objective CMA-ES



	Evolutionary Strategy	Genetic Algorithm	Bayesian Optimization
1. Estimate Distribution	Search distribution by e.g. Normal	Search distribution = population	Estimate f(x), and uncertainty thereof
2. Choose x	Sample from distribution	Sample from population, with cross-over	Sample x with e.g. max expected improvement*



*Snoek, Larochelle, Adams. "Practical Bayesian Optimization of ML Algo", NIPS2012

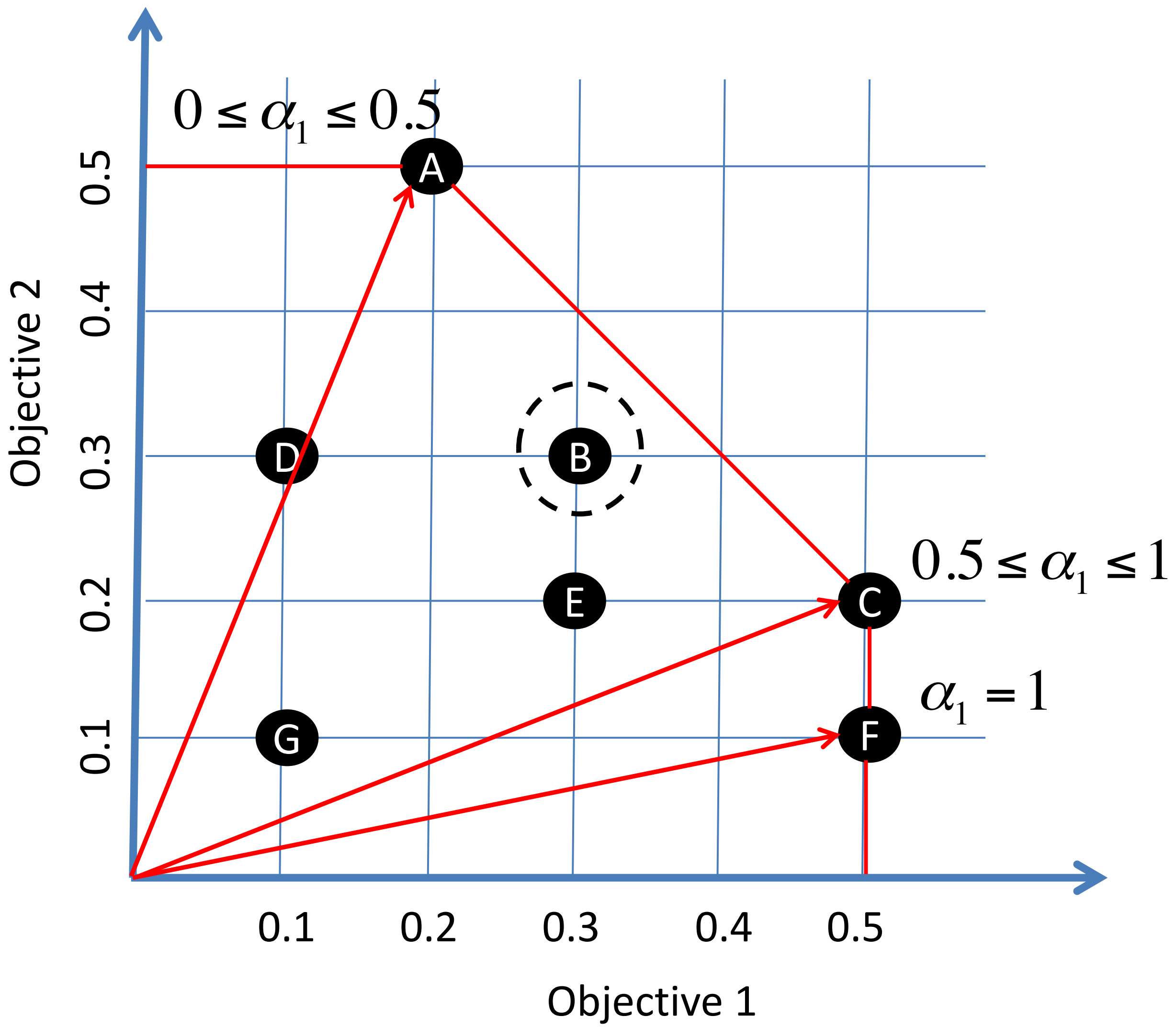
Aside: Alternative to Pareto Optimality

- Combine multiple objectives into one

$$\max_x [f_1(x), f_2(x), \ldots, f_M(x)]$$

$$\text{Scalarization: } \max_x [\sum_m \alpha_m f_m(x)] \qquad \alpha_m \geq 0, \sum_{m=1}^{M} \alpha_m = 1$$

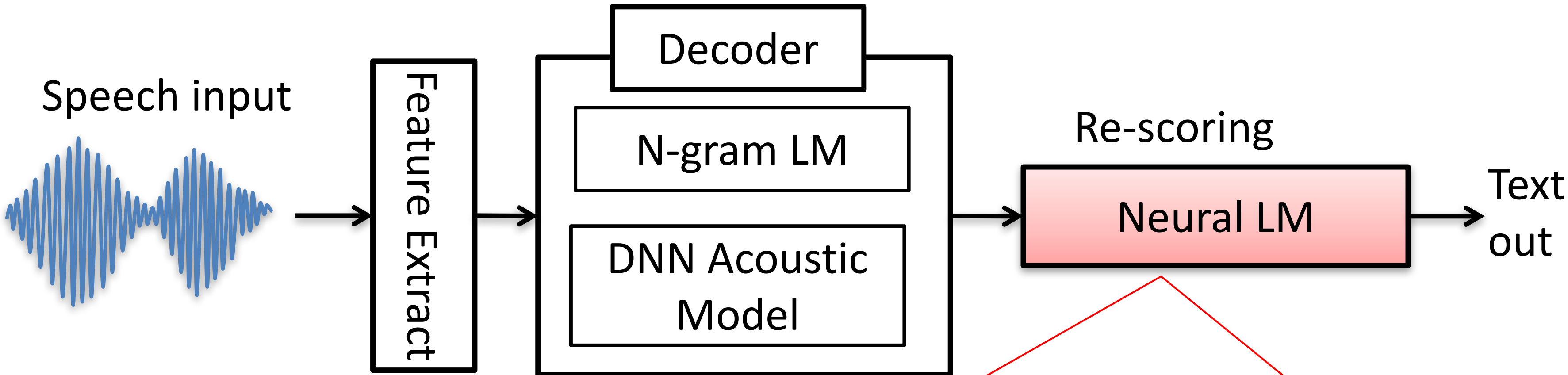
Pareto vs. Scalarization
Pareto points not on Convex Hull are missed



Outline

1. Motivation
2. Problem Definition
3. Multi-objective evolution strategy
4. Experiment on speech recognition
5. Ongoing work

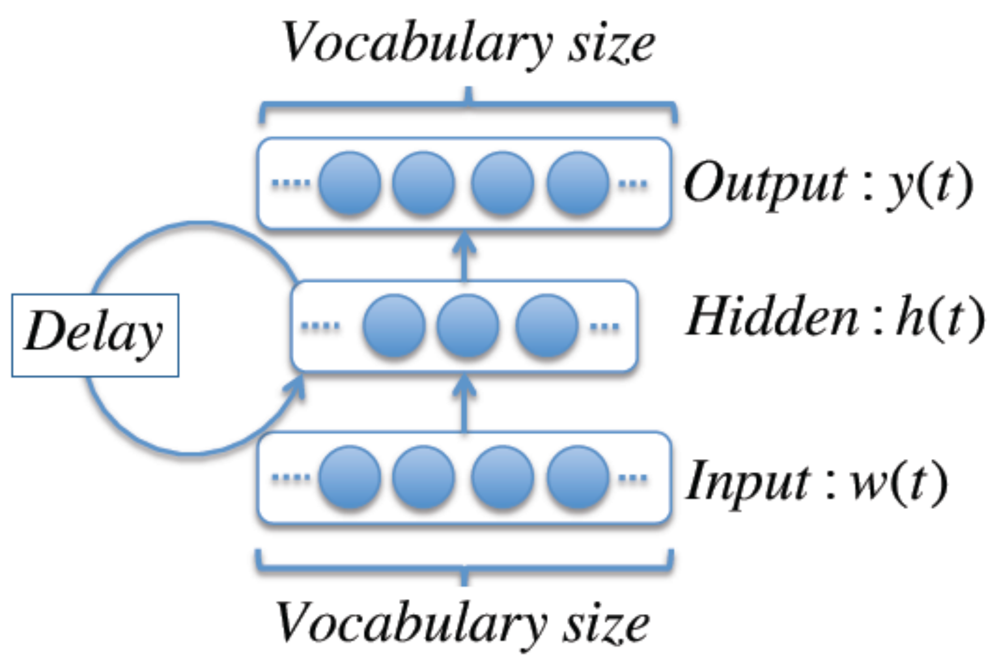
Setup 1: Speech Recognition N-best re-scoring



2 Objectives
- Word error rate
- Training time

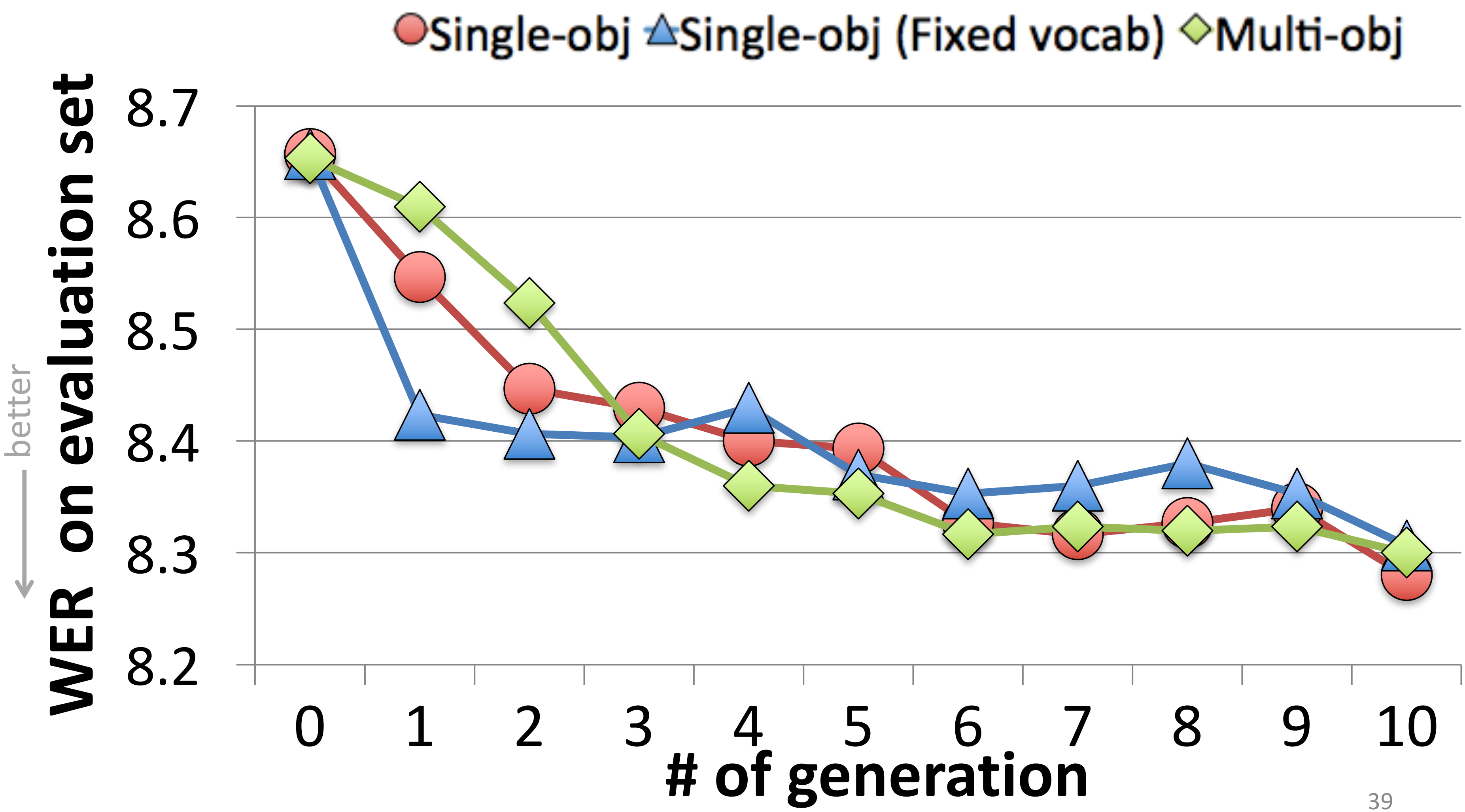
37 Hyperparameter Types
Population: 30

Trained on CSJ data

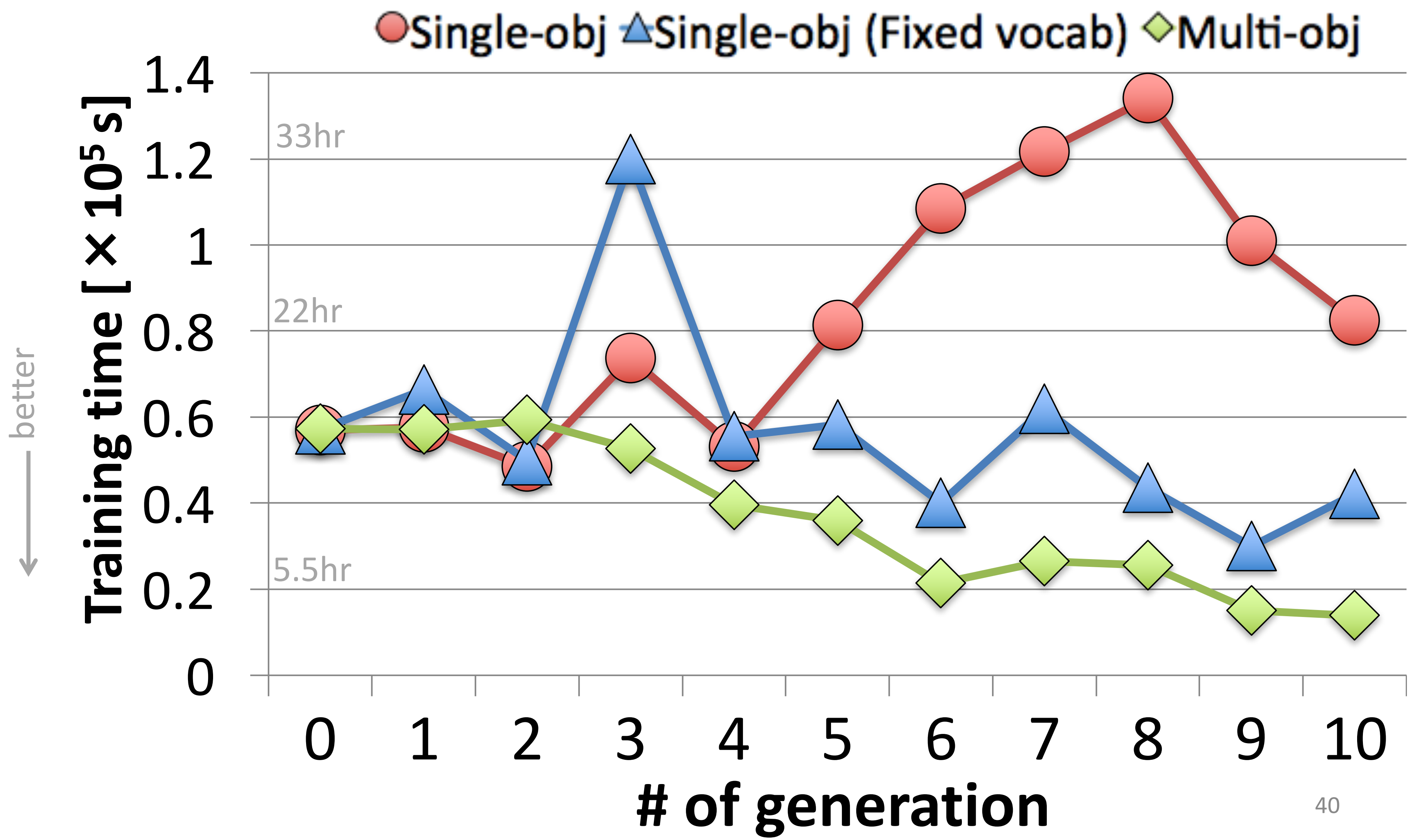


Category	Hyper-parameters (x)	Initial value
Structure	# hidden layers (1-10)	2
	# units in each hidden layer	300
	# units in word embedding	300
	vocabulary size	10000
	unit type in each hidden layer (LSTM, RNN, FF)	LSTM
Training	minibatch size	32
	initial leaning rate	1
	learning rate decay	0.5
	decay start epoch	6
	dropout ratio	0.5
	momentum	1E10
	gradient clip	5
	initial forget gate bias	1
	optimizer type (SGD, ADAM, ADADELTA, RMSprop)	SGD
	meta-parameters in optimizers	-
Scoring	NN-LM weight (interpolate with n-gram)	0.5
	acoustic weight	14

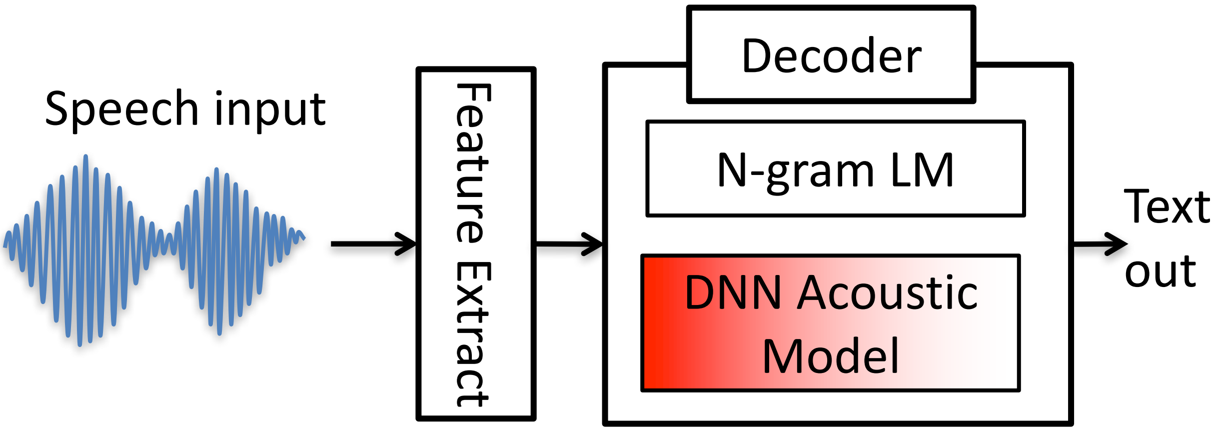
Word Error Rate (WER) improvement each generation



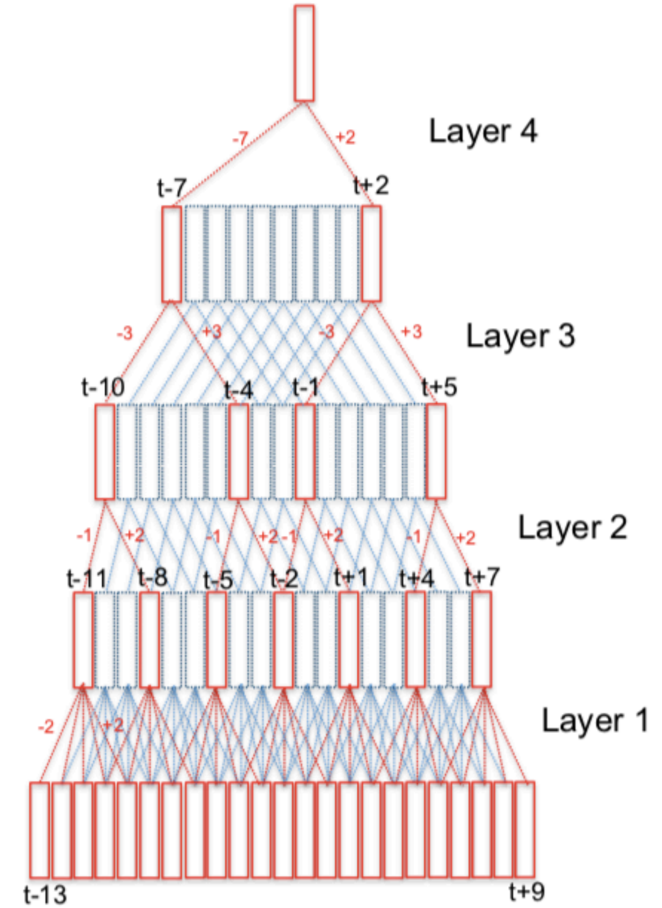
Training time differences between single and multiple objective evolution



Setup 2: Speech Recognition acoustic modeling



Chain TDNN: Peddinti, et. al. (2015)
A time delay neural net for efficient modeling of long temporal contexts



2 Objectives:

- Word error rate
- Model Size

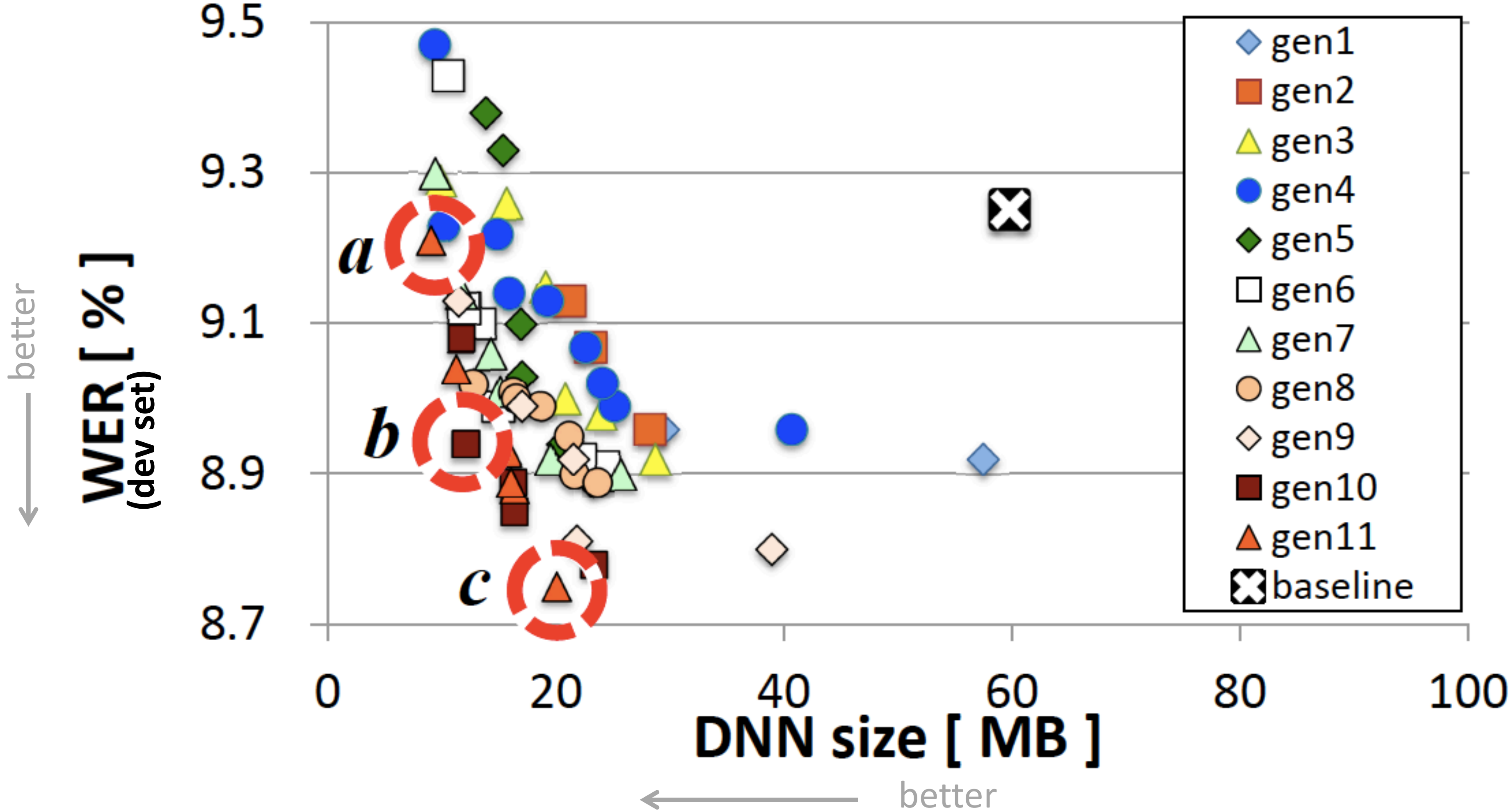
7 Hyperparameter Types for Time-Delay Neural Net

e.g. #layer, #unit, learn-rate

Population: 30

Trained on CSJ data

Pareto points in each generation

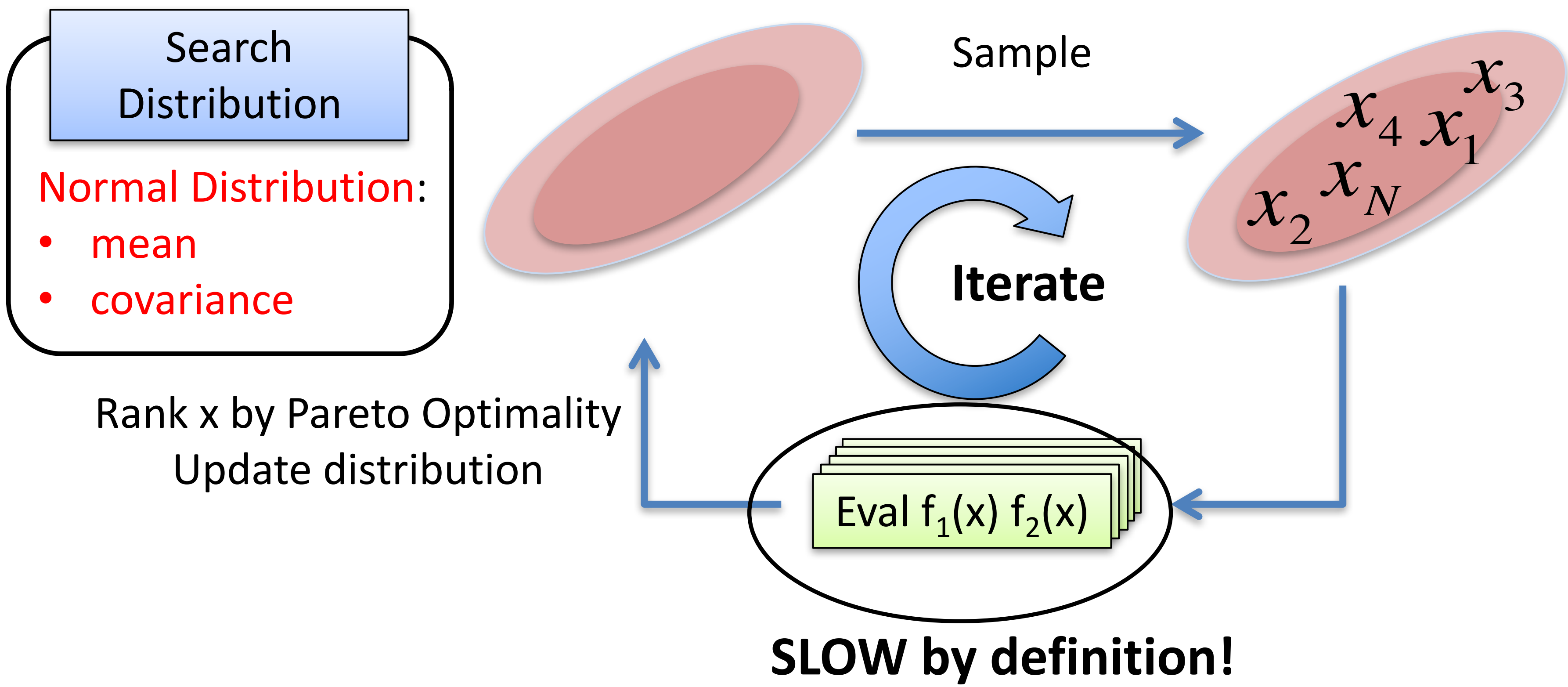


In a realistic use case: give human the Pareto frontier to decide what to deploy

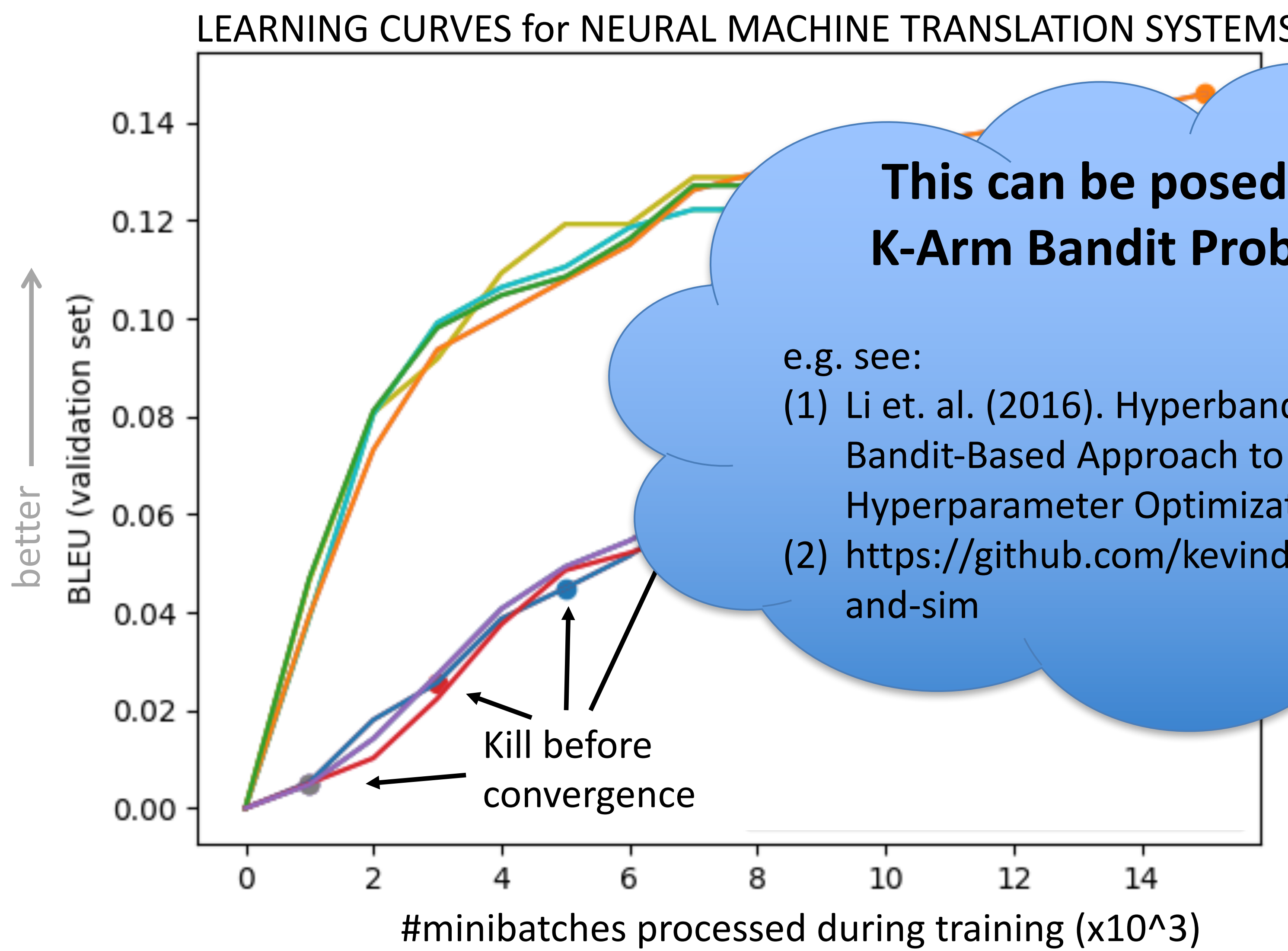
Outline

1. Motivation
2. Problem Definition
3. Multi-objective evolutionary strategy
4. Experiment on speech recognition
5. Ongoing work

1. Speeding-up the Black Box



Simple Idea (inspired by *graduate student descent*): "Kill the training job when it looks hopeless"



This can be posed as a K-Arm Bandit Problem!

e.g. see:

(1) Li et. al. (2016). Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization
(2) https://github.com/kevinduh/hyperband-sim

2. Building Benchmark Datasets

- Currently difficult to compare hyperparameter optimization methods due to computational resource bottlenecks
- Solution: create ***reusable*** benchmarks

1. **Train MANY models on some dataset beforehand**
2. **Publish all (x,y) as a table**
3. **Benchmark methods on a finite universe**

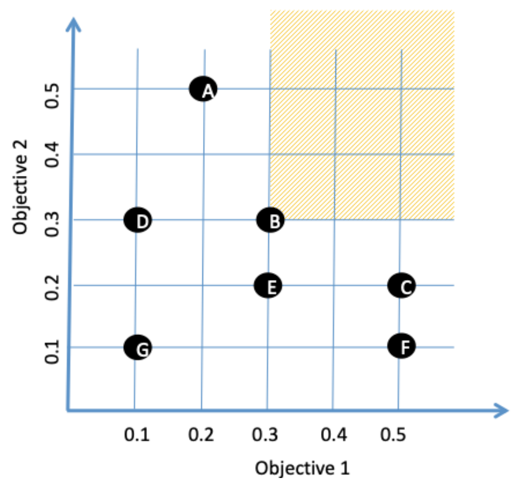
If you have 500+ models on some dataset lying around, let me know!

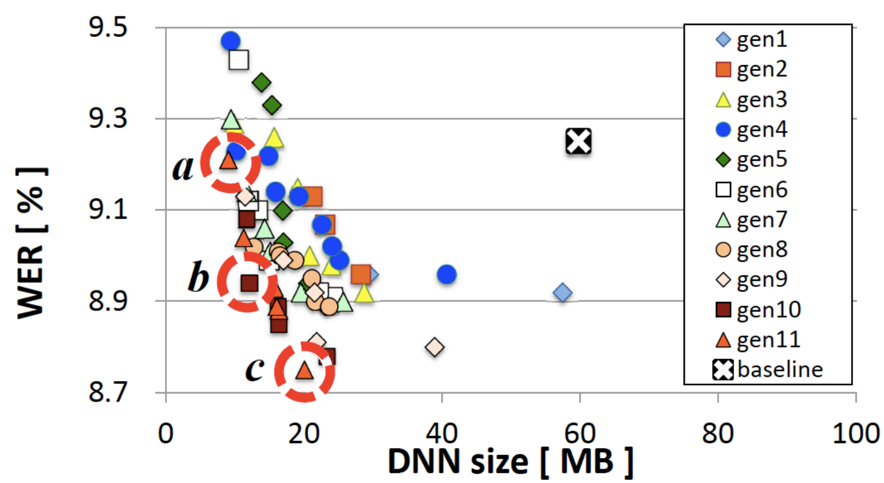
Summary

1. Hyperparameter Optimization is needed for scalable development of DNNs

Find $x^{*}=\text{argmax}_{x}\ f(x)$ with few evaluations

2. Multi-objective methods viable with Pareto





3. Fast & Accurate is (sometimes) achievable!