**Real-valued Optimization by Subspace Projection and   
Multi-armed Bandit Techniques**

1. **Introduction**

Background:

Describe the common dilemma between ***exploration*** and ***exploitation*** for real-valued optimization algorithms.

Describe our ***basic assumption*** that problems worth solving are ***composed of observable uni-modals(?)***. Quote Goldberg or no free lunch theorem?

Hie

Purpose:

We propose a technique that helps ***identify areas*** to explore and ***allocate resources*** according to remaining evaluations.

**Identify areas (region of interests?):**

Describe how subspace projection creates a ***well-defined boundary*** that some algorithms need.

Describe how subspace projection helps ~~to~~ solve ***inseparable problems*** while ***enhancing the probability*** for finding optimum.

**Resource Allocation:**

Describe different strategies one should take given different **evaluations left**.

Describe how ***multi-armed bandit*** algorithms are ***suitable*** for our scenario: learning model from outcomes while actions do not change the state of the world. (instead of decision theory, reinforcement learning and markov decision)

Roadmap:

Following sections introduces related subjects: optimization algorithms, Linear Projection, Clustering, multi-armed bandit algorithms.

Then we present our new bandit techniques

Finally, we present experiment results and give conclusion

1. **Real-valued Optimization Algorithms**

**2.1 Covariance Matrix Adaptation Evolution Strategy**

Describe history of **Evolutionary Strategies** ((1+1)ES -> CSA-ES -> CMA-ES)

Describe the underlying covariance matrix modal

Describe how to update **mean**

Describe how to update **covariance matrix**

Describe **step-size** control

Give pseudo code

**2.2 Standard Particle Swarm Optimization**

Briefly describe **swarm intelligence** and history of PSO

The **purpose** of SPSO: Well-defined principals & milestone for comparison

Describe **swarm size** definition and **basic elements** for each particle

Describe **velocity update** for SPSO2006 SPSO2011

Describe **random topology** and when to **update topology**

Describe **boundary** and out-of-bound handling

Give pseudo code

**2.3 Ant Colony Optimization for Continuous Domain**

Describe history of **ant based algorithms** and ants foraging behavior

Describe **discrete probabilistic pheromone model** for Ant System

Describe **continuous probability density function model** for ACOR

Describe the **Gaussian kernel** and the **Solution archive**

Describe how **generate solution** from Gaussian kernel and **archive update**

Give pseudo code

1. **Linear Projection**

**3.1 Affine Transformation**

Describe Translation, Rotation, Scaling, Shearing with 2D-matrix example

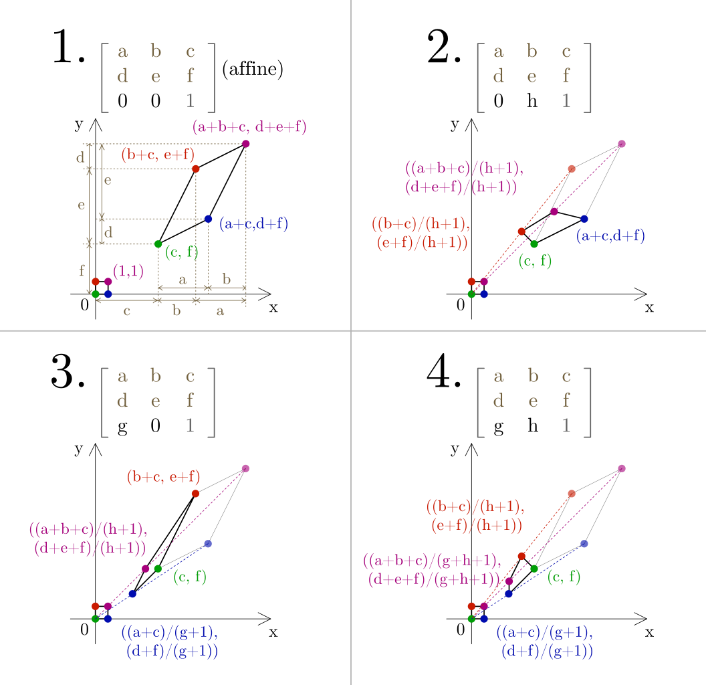
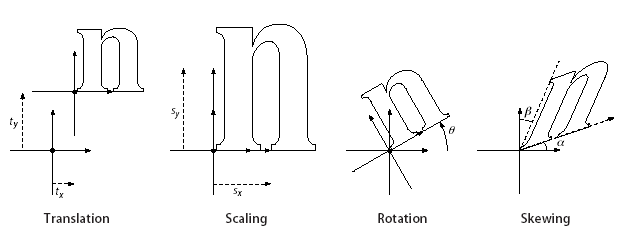
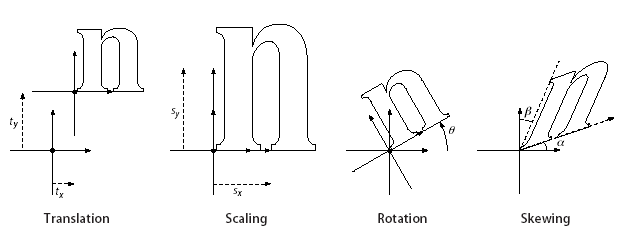
**3.2 Linear Projective**

Describe **homogeneous coordinate**

Describe **basic projective** transformation

Describe **inverse projection** and **limits**

Describe how a (D+1)^2 matrix transforms a D-dim point to **subspace**



**3.3 Optimization for Transformation Matrix**

Describe our **loss function**:  
 Points within boundaries should be in

Points from other Clusters should be out

Best point should be at the center

We hope to get a Normal distribution with std = 0.2?

Minimal error for inverse transformation

Describe the difficulty of finding a **fast** yet **powerful** optimization problem for **hyperparameters** **optimization** (CMA-ES -> (1+1)ES)

Describe **(1+1)-ES** and give pseudo code

1. **Clustering Techniques**

**4.1 K-means clustering with silhouette score**

Describe how K-means clustering works and why it is popular

Describe how silhouette score decides number of clusters

Describe that K-means cannot identify density nor unimodality

**4.2 Gap statistics**

Describe how gap-statistics estimates number of clusters

**4.3 Skinny-Dip**

Describe how Dip-test checks unimodality

Describe skinny-dip clustering

MDL Minimum Description Length

BIC Bayesian Information Criterion

1. **Multi-armed bandit Algorithms**

**5.1 The Multi-armed Bandit Problem**

Agent needs to decide in K arms to pull at time t and receives a reward.

Describe **policy** and **regret**, goal is to **minimize regret**

**5.2 Common Bandit Algorithms**

Multi-Armed Bandit Algorithms and Empirical Evaluation

5.2.1 Epsilon

5.2.2 UCB

**5.3 The new Bandit Techniques**

Not minimize regret, but maximize the probability to get rank 1

1. **(The new Bandit Techniques)**

**6.1 Framework of the New Bandit Algorithm**

Describe the goal: search for area and allocate resources

Each algorithm needs to be modified to satisfy following conditions:

1. Update one individual at a time
2. Transform to a subspace given a matrix
3. Replace one individual with a given position and fitness

Give pseudo code

**6.2 Initialization and Clustering**

Initialize with 100\*D points and keep only top 50% for clustering

Iteratively add 10\*D points until estimated cluster number is the same

With a given cluster number k, do K-Means clustering

**6.3 Arm Generation and Subspace Projection**

Each arm is composed of an algorithm and a projection matrix

Initial matrix is set as a tight hyperbox that contains all points in cluster

Optimize matrix one by one

Resize each cluster to match algorithm population

Start algorithm with given points

**6.4 Bandit Technique for Remain Evaluations Allocation**

Calculate remain evaluations allocation and normalize to 0~1

Add newly calculated allocation to record

Chose arm with argmax allocation to pull (update one particle)

Max arm Allocation - 1

**6.5 Reclustering**

If distance of best to center is greater than 0.3, recluster

1. **Experiments**

**7.1 Test Problems (CEC2005 benchmark problems)**

Describe termination criterion and evaluation method for CEC2005

Describe 25 repeat runs and

**7.2 Experiment Settings & Results**

Describe the importance for identical initialization for real-valued problems

Describes the parameters setting for CMA-ES, SPSO and ACOR

Describe our bandit parameters setting: init\_pop. Max\_arms, 1+1 initStep…

**7.3 Results**

Give table of 1st, 7th, 13th, 19th 25th error at FE = 1e3, 1e4, 1e5

Give medium run error figure

**7.4 Discussion**

1. **Conclusion**

Contribution:

find potential region to search

allocate resources

Weakness:

More NFE on unimodal and moving region

Need for a good clustering technique that identifies underlying uni-modal

Future work:

Non-linear transformation

Better clustering techniques for better uni-modal performance