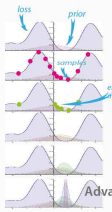
Benchmarking Cross Entropy Method for partially Decomposable Problems

A Batch sampling method for large-scaled structured optimization

Siamak Ravanbakhsh, Russell Greiner
Computing Science Department, University of Alberta

Batch Sampling for Optimization

Monte Carlo methods for optimization includes both sequential and bach sampling methods:
Simulated annealing is a sequential method.
Cross-Entropy and Probability Collectives are a batch sampling method



Cross Entropy Method

Start from a prior
Repeat until convergence
- Take samples from current distribution
- Calculate the loss for samples
- Select top (Elite) samples
- Find maximum Likelihood for elites

Advantages of batch to sequenial methods

Highly paralelizable
Calculation of loss for each instance is independent of the other instance and may be paralelized.
Availability of Sensitivity Information
Peakiness/flatness of the distribution gives us extra information. value of the optimization function. \
Our Alg. (CEED) uses Fisher Information to extract this information.

Partial Decomposibility

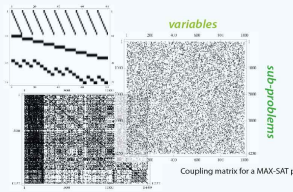
The basic idea is to use sensitivity information that is available from batch sampling, to simultanuously optimize several (sub)-problems that share some variables.

- Consider problem of the form: $L(\vec{x}) = \sum_{p=1}^{P} L_p(\vec{x}^p)$ when $\vec{X}^p \subseteq \vec{X}$
- We call such problems ***partially decomposible***.
- CEED is designed to exploit such structures in the loss.

Coupling matrix representation

Each row represents a sub-problem and each column a variable.
Black squares show when a variable participates in a sub-problem

Coupling matrix for 3x3 Sudoku problem



Coupling matrix for a MAX-SAT problem

Coupling Matrix for a large curve-fitting problem
(for analysis of Nuclear Magnetic Resonance Spectrum)

CEED Method

Start from a Prior *(with a product form)*
Repeat until convergence

The same sub-routine

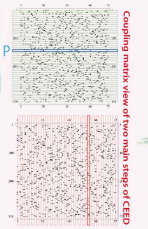
For each sub-problem P
- Consider the marginal of prior over the vars in P
- Take samples from this marginal
- Calculate the loss for all samples
- Select top (elite) samples
- Calculate Maximum Likelihood dist. *(with a product form)* for these samples

For each Variable V
- Consider all sub-problems that have V in their domain
- Consider their posterior distribution over V
- **Combine** these distributions to get a unified posterior

Use joint distribution of unified posteriors as the next prior

Return the mode of the last posterior as the optimal value found

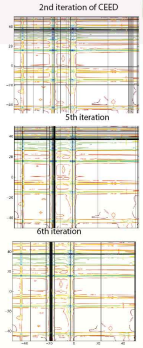
Use Fisher Information of a distribution to linearly combine the estimated max-likelihood parameters.
For discrete distributions this conicides with combination using inverse of variance-covariance matrix.
For general distributions variance of the ML estimates is not readily available, however Fisher Information is shown to asymptotically converge to this value for ML estimator (Cramer-Rao theorem)
One might also combine distributions by finding a distribution with minimum sum of KL-divergence to all given dists.



Foveating for Continuous Domains

- We consider adaptive discretization of continuous domain.
  At each iteration, space is discretized into bins such that in the beginning of each iteration all bins have the same probability.
- Samples are taken uniformly from all bins and also uniformly within each bin
- Black lines show the boundary of bins for two variables.

Here the black lines are the borders of different bins. Vertical lines show the bins for the first var and horizontal lines are binning the second var.



Contur plot of the Optimization Landscape and convergence of CEED by foveating

Benchmark Problem

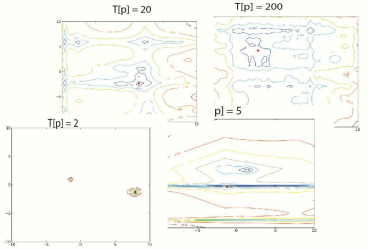
We designed a partially decomposible multi-extremal loss, such that we can change
- the number of variables ($|\vec{x}| = V$)
- the number of sub-problems (P)
- the number of variables in each sub-problem ($|\vec{x}^p| = V[p]$)
- the difficulty of each sub-problem-- i.e. number of terms (T[p])

$$\text{problem}: \quad L(\vec{x}) = \sum_{p=1}^{P} L_p(\vec{x}^p) \qquad \vec{X}^p \subseteq \vec{X}$$

$$\text{sub-problem}: \quad L_p(\vec{x}^p) = \sum_{t=1}^{T[p]} c_t L_{p,t}(\vec{x}^{p,t}) \qquad \vec{X}^{p,t} \subseteq \vec{X}^p$$

$$\text{terms}: \quad L_{p,t}(\vec{x}^{p,t}) = e^{\sum_{x_j \in \vec{x}^{p,t}} \{w_{p,t,j}(x_j - s_{p,t,j})\}}$$

Optimization Landscape for various difficulties (V = 2)

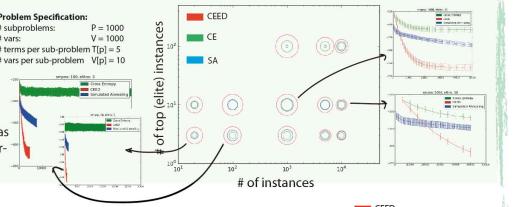


CEED against CE and Simulated Annealing (SA)

Side Note:
For our experiments we used Lester Ingber's Adaptive Simulated Annealing. We set the correspondence between the parameters of CEED and CEED with that of SA as follows:
# of instances corresponds to number of instances per temperature for SA.
The ratio of elite instances to total samples corresponds to the Boltzmann constant in SA.

These figures represent the average convergence rate corresponding to different parameters for all algorithms. Notice that larger circle corresponds to faster and better convergence.

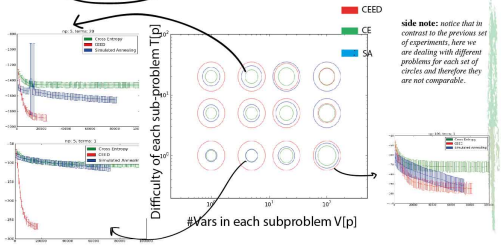
Problem Specification:
subproblems: P = 1000
vars: V = 1000
terms per sub-problem T[p] = 5
vars per sub-problem V[p] = 10



Conclusion: CEED using as few as 20 instances per iteration outperforms CE when using 5000 instances (The corresponding circle is larger).

Sample size & Alg Parameters

Problem Specification:
subproblems: P = 1000
vars: V = 1000
instances per iteration 1000
elites instances 100

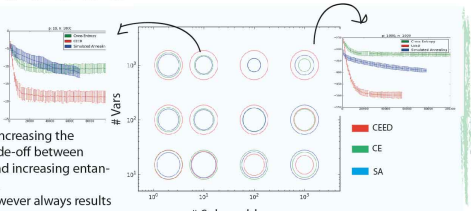


side note: *notice that in contrast to the previous set of experiments, here we are dealing with different problems for each set of circles and therefore they are not comparable.*

Conclusion: CEED is most advantageous when the number of variable in each sub-problem is small (i.e. the problem is more decomposible) and/or sub-problems are relatively easier to solve.

Difficulty of Sub-problems & Vars in each Sub-problem

Problem Specification:
terms per sub-problem T[p] = 5
vars per sub-problem V[p] = 10
instances per iteration 1000
elite instances 100



Conclusion: Results suggests that by increasing the number of sub-problems there is a trade-off between having more sources of information and increasing entanglement (decreasing decomposibility).
Increasing the number of variables however always results in less coupling between sub-problems and is always advantageous.

Number of Vars & Sub-problems

Future work

Energy minimization in molecular mechanics:
Any energy minimization task in molecular mechanics involves minimization of sum of many energy terms, therefore the partial decomposibility structure. Each energy term usually involves only a subset of variables. For example energy terms corresponding to tortion angle in protein folding only depends on the configuration of four related residuals.
This motivates us to apply CEED to ab-initio protein folding problem or to similar but less explored problem of protein-docking.

Energy minimization in MRF:
In many problems of vision such as streo-matching, denoising, image segmentation etc. the task is to find a labeling that minimizes an energy function. This energy function may be written as a sum of two terms for each individual label. One term penalizes deviation from the prior and the second penilizes distance from the observation. Again the problem demonstrates a decomposible structure and we inted to apply CEED to such problems.