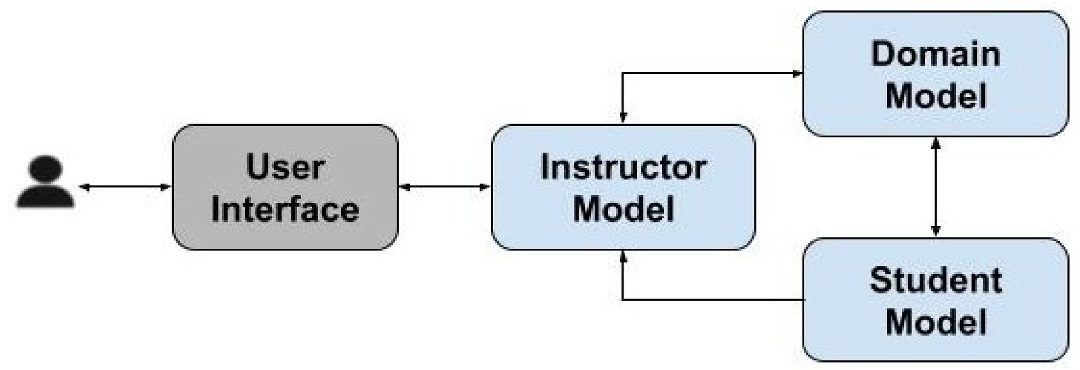
# A Bandits Approach to Intelligent Tutoring Systems using Concept Evolution Estimation

Sudha S, Arun Rajkumar

Indian Institute of Technology Madras

## Introduction

Conventional teaching methods take a one-size-fits-all approach – same course contents, evaluation methods, resources, etc.




Each learner is different, has different abilities and requirements. An Intelligent Tutoring System (ITS) tailors the course contents to each student.

**Important Modules** include User Interface, Domain Model, Student Model and the Instruction Model.

- **Domain Model** deals with *what* is being taught
- **Student Model** deals with *who* is being taught
- **Instruction Model** deals with *how* to teach

## Related Work

There are many works in the field of ITS. We have listed below a few important works and the approach they take.

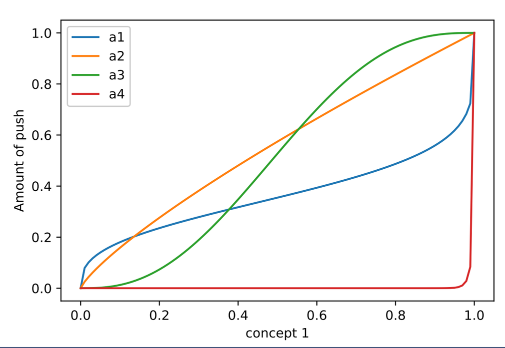
- **[1][Lan, Biraniuk]** propose two UCB based algorithm CLUB & ACLUB which fix the sequence of assessments & insert a learning action in between the assessments. Their aim here to maximize the learner's score in the immediate next assessment.
- **[2][Manickam et al.]** work with the same setting as above. The propose an algorithm based on Thompson Sampling & Knowledge Gradients.
- **[3][Wang, et al.]** take a POMDP approach to ITS where they use the questions asked by learners as observations & the answers the ITS provides as actions. However, they only allow $\{0, 1\}$ states concept values.

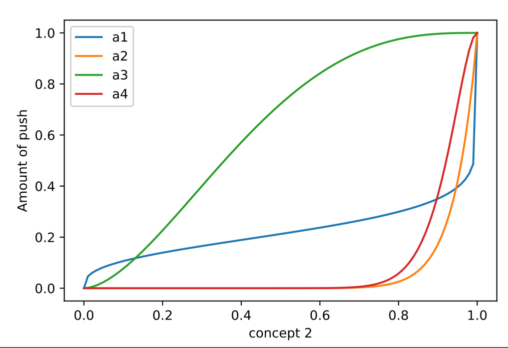
Most of the work mentioned above take a divided approach to it. Our aim to take a holistic approach to ITS.

## Problem Setting & Assumptions

There are $N$ learners & $K$ concepts.

- **Concept Vectors** is in $C_i \in [0,1]^K$ where $C_{ij} \in [0,1]$
- **Learning Action** could be anything such as Video lecture, book chapters, etc. Assumed to give Beta Distributed learning push.
- **Markovian Evolution**: The learning level of a learner depends on his/her current knowledge level & the Learning Action chosen at current step.
- **Noisy Feedback**: We assume we have at our disposal a noisy version of the learner's concept vectors.




## Method Proposed – Independent Concepts

We study two cases of the problem: **Independent Concepts** and the **Dependent Concepts**

In the independent case, there are no dependencies among the concepts. i.e., each concept could be taught irrespective of the learner's knowledge about other concepts.

**Update Equation**

$$c_i^{t+1} = c_i^t + \text{Beta}(\alpha_a, \beta_a, c_i^t) \cdot (1 - c_i^t) \quad (1)$$

**Parameters to Estimate** here are the Beta Distribution parameters $\alpha$ & $\beta$ associated with each action. And there are $2K$ such parameters for each Action.

## Method Proposed - Dependent Concepts

In the dependent concepts case, we assume a pre-requisite relationship among the concepts. Hence, before teaching a concept, its pre-requisite concepts need to be taught.

**Update Equation**

$$c_i^{t+1} = c_i^t + \sum_{j=1}^{D} c_j^t \lambda_{j->i} \cdot \text{Beta}(\alpha_a, \beta_a, c_i^t) \cdot (1 - c_i^t) \quad (2)$$

where $D$ is the number of prerequisite concepts to $c_i$ and

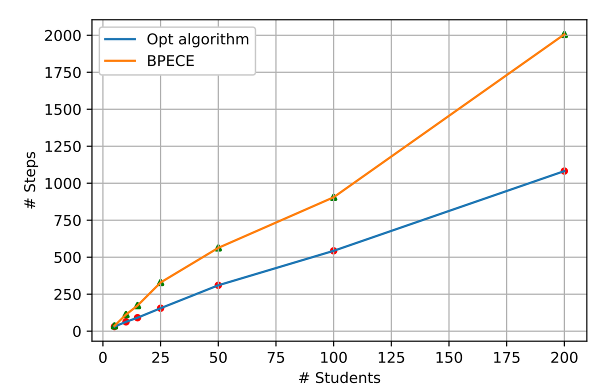
$$\sum_{j=1}^{D} \lambda_{j->i} = 1 \quad (3)$$

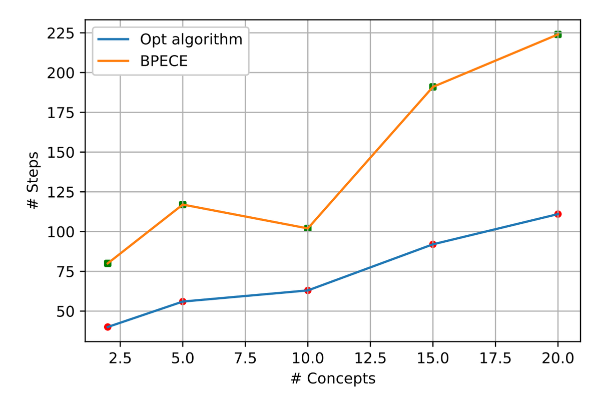
Because of the update equation, our objective function becomes:

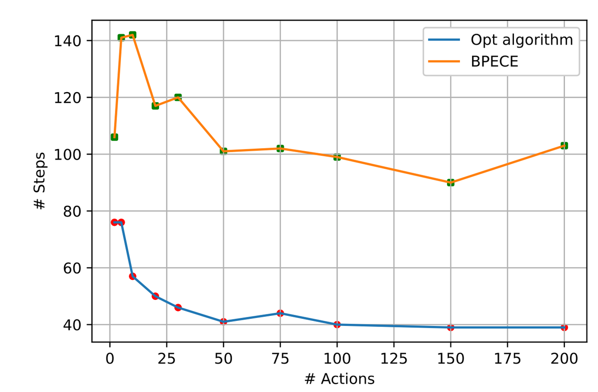
$$f(\alpha_a, \beta_a) = \left(\frac{c_i^{t+1} - c_i^t}{1 - c_i^t}\right) - \text{Beta}(\alpha_a, \beta_a, c_i^t) \quad (4)$$

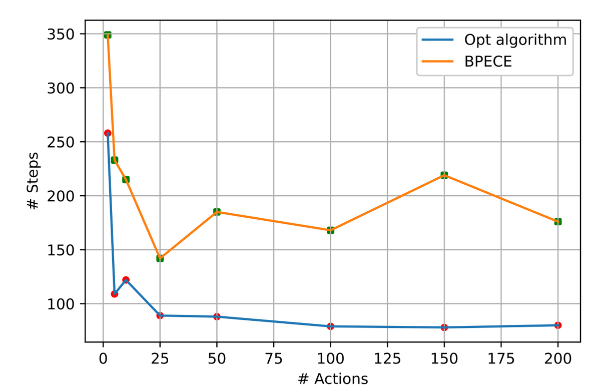
**Parameters to Estimate** here are not just he Action Beta parameters, but also the lambda parameters involved.
We propose the Bandits based **Parameter Estimation for Concept Evolution** (BPECE) algorithm as shown in Algorithm 1.

## Results



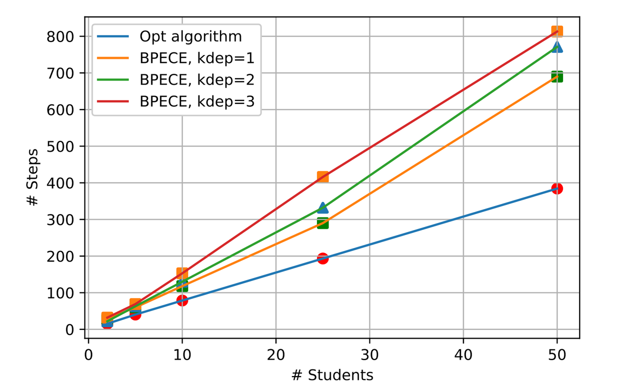



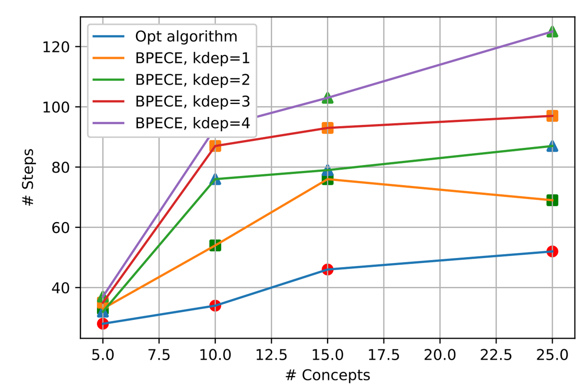

## Algorithm: BPECE

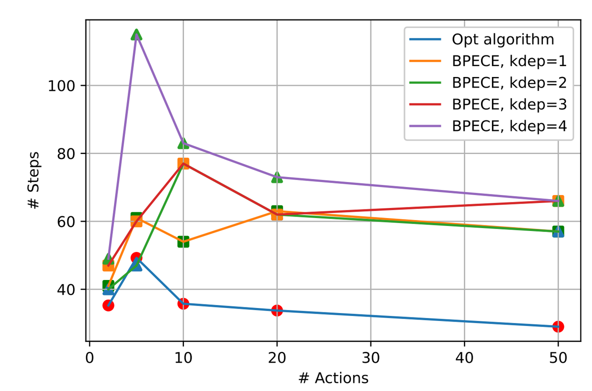
**Input:** A set of learner concept vector estimates, $C_j, j = 1, 2, \ldots N$
Parameters $A_{min}, D_{min}, \epsilon$
**Output:** Next action $a_j$ for each learner $j$
**for** $j \leftarrow 1$ *to* $N$ **do**
  **if** $\exists a \in A$ *where* $count(a) < A_{min}$ **then**
    $a_j \leftarrow a$
  **end**
  **else**
    **for** $c_{ji} \in C_j$ **do**
      **if** $c_{ji}$ *is Independent* **then**
        Estimate the $(\alpha_a, \beta_a) \forall a \in A$ using Zeroth-Order Optimization on 4
      **end**
      **if** $c_{ji}$ *is Dependent* **then**
        Initialize $\lambda_{k->ji}$ values uniformly $\forall (k, ij)$
        **while** $\lambda_{k->ji}$ *AND* $(\alpha_a, \beta_a) \forall a \in A$ *are not converged* **do**
          Fix $\lambda_{k->ji}$
          Estimate $(\alpha_a, \beta_a) \forall a \in A$ using Zeroth-Order Optimization on 4
          Fix $(\alpha_a, \beta_a) \forall a \in A$
          Estimate $\lambda_{k->ji}$ using the Neural Nets
        **end**
      **end**
    **end**
    Update $C'_{j_a}$ using Equation 1 & 2 $\forall a \in A$
    **With** probability $1 - \epsilon$
    $a_j \leftarrow \arg\max_{a \in A} ||C'_{j_a} - C_j||_2$
    **With** probability $\epsilon$
    $a_j \leftarrow$ choose an action $a \in A$ uniformly at random
  **end**
**end**

**Algorithm 1:** BPECE

## Results




## References

[1] Andrew S Lan and Richard G Baraniuk.
A contextual bandits framework for personalized learning action selection.
In *EDM*, pages 424–429, 2016.



cs19s017@cse.iitm.ac.in, arunr@cse.iitm.ac.in