COSC 419: Mobile Educational Game Development

Dr. Bowen Hui
University of British Columbia Okanagan

Recall: AI Techniques in Games

- Most common is for controlling non-player characters (NPCs)
- Most common techniques used:
  - Finite state machine (FSM)
  - Search Tree

Simple AI techniques
- Relatively predictable behaviour
- Requires min. resources

- Niche area: uses machine learning to adapt behaviour throughout game play
  - Good for developing relationship with player
  - E.g. Petz

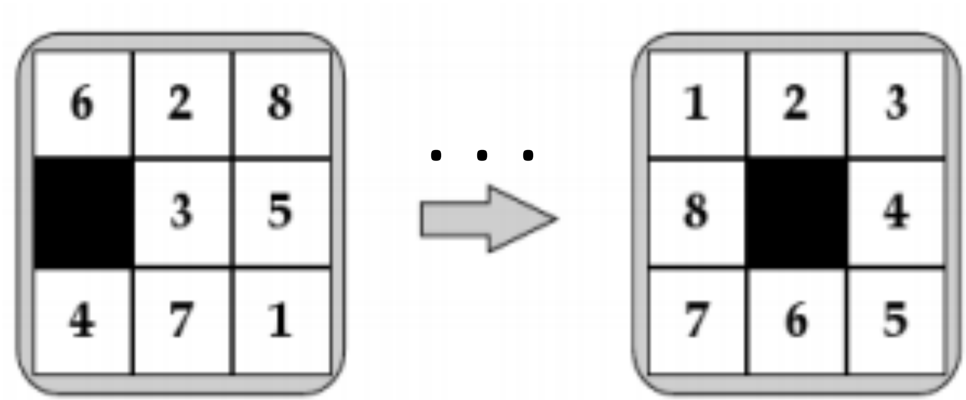
What is a Search Tree

- A tree data structure
- Reduces the problem of finding next move to a graph search
  - Define set of possible states
  - Define possible actions at each state
  - Define goal state (or utility for each end state)
- Brute force search
- Allows look-ahead into the future

Deep Blue beats Kasparav (1997)

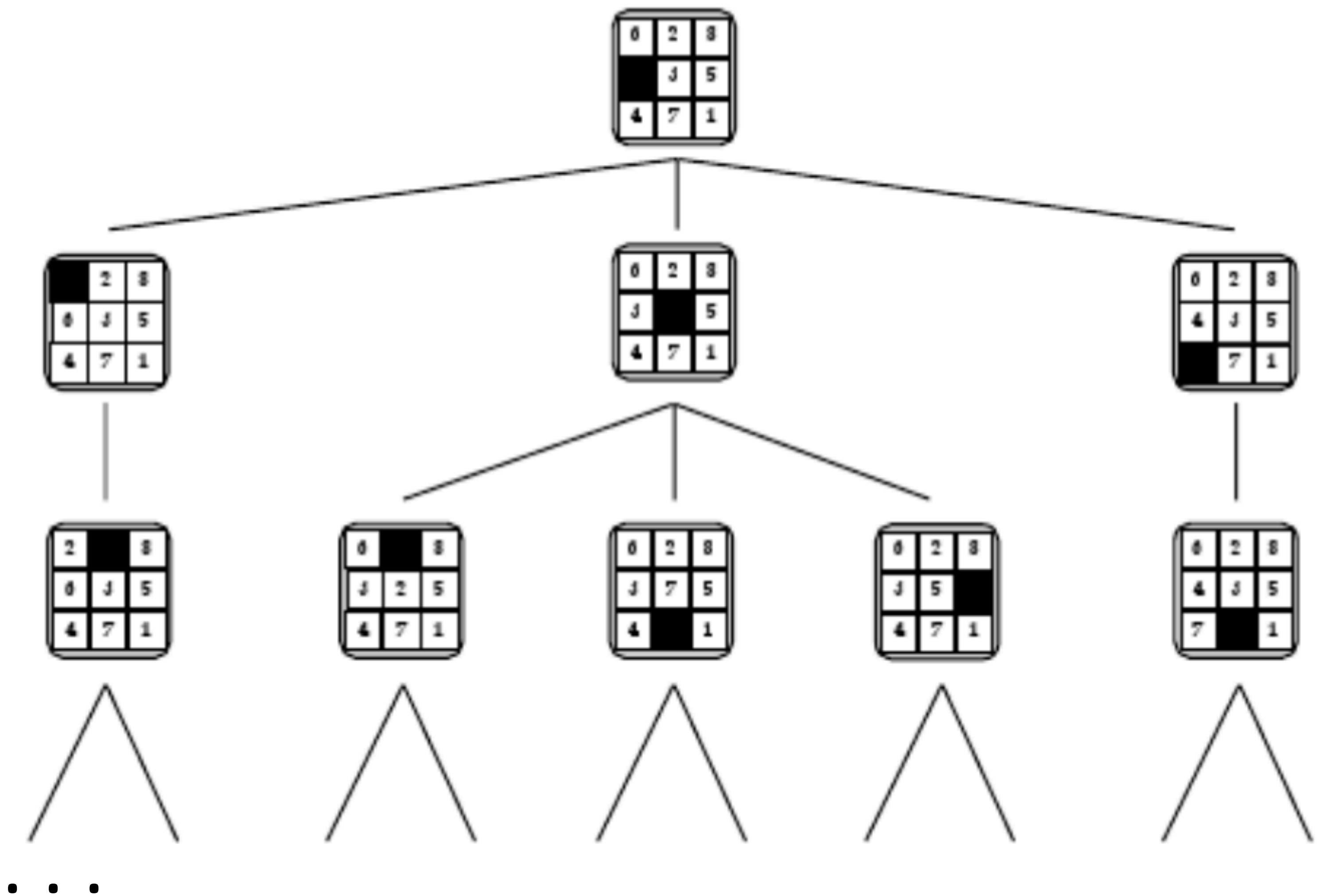
Game of Eight Example

- Move squares from initial configuration to (eventually) the final configuration



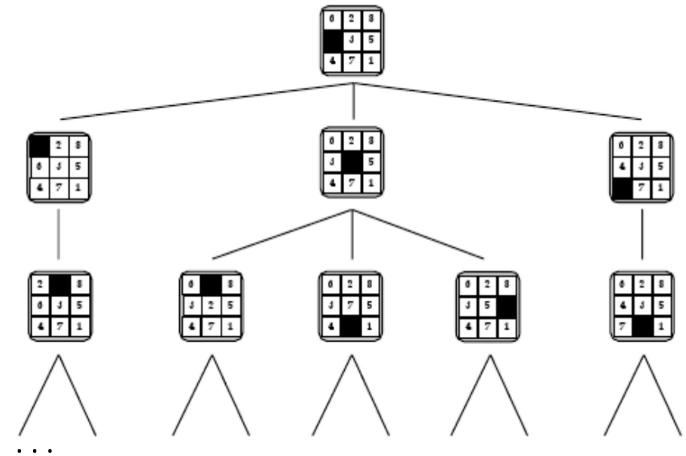
- Possible actions: move any neighbouring numbered square into empty square (black)
- Each state is a resulting configuration after one move

Game of Eight Game Tree



Game of Eight Game Tree

- Enumerate all action-state combinations until goal state reached

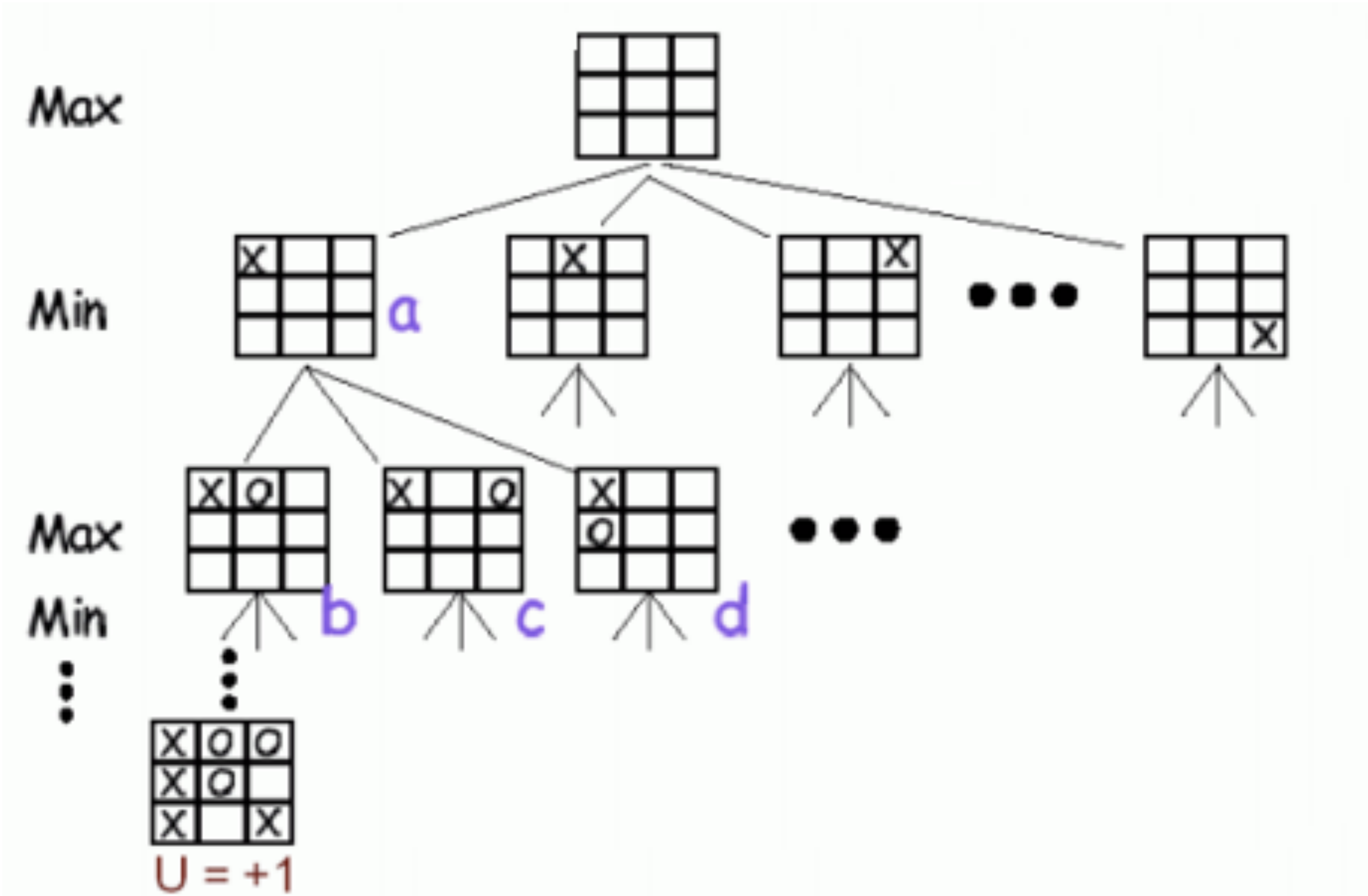


- Take (shortest) path that goes from initial state to goal state
- Illustrates the brute force nature of the solution

Computational Constraints

- Action space:
  - Number of possible moves per state
  - Tree has more branches if more possible actions
  - E.g. Two empty squares
- State space:
  - Number of possible states
  - Tree has more depth if more possible states
  - E.g. Game of 16
- More complicated when multiple players are involved

Tic Tac Toe Example



Search Algorithms

- Various algorithms developed to find best way to get to a winning goal state
- Uninformed search:
  - Blind search
  - Focus on computational bounds for each strategy
- Informed search:
  - Directs search to expand less costly nodes
  - Applies heuristic function
  - Focus on best case results in the general case

Adapting to Player Interaction

- E.g. game to raise pet
  - Good for developing relationship with player
  - More nurturing leads to more loving pet
  - Pet learns when it can play with you
- Other opportunities?
  - Game difficulty
    - Game play: speed, number of enemies, available weapons, etc.
    - Question to solve (e.g. “4 x 5” vs. “32 x 78”)

Motivation

- In 2012, PHYS 112 @ UBCO had:
  - 32% students failed
  - 79% students got < 80%
- Typical in first year STEM courses
  - Explore intelligent tutoring systems as innovative learning strategy
- Third year URA project by Matt Bojey
  - Experience as Physics TA
  - Computer Science & Math Honours student

KRIT: Kirchhoff's Rules Intelligent Tutor

- Aimed at teaching PHYS 112 students about Kirchhoff's Rules
  - Basic understanding of the rules
  - Application of the rules
  - Creating new problems
- Project focus
  - Design and implementation
  - Evaluation with Physics students

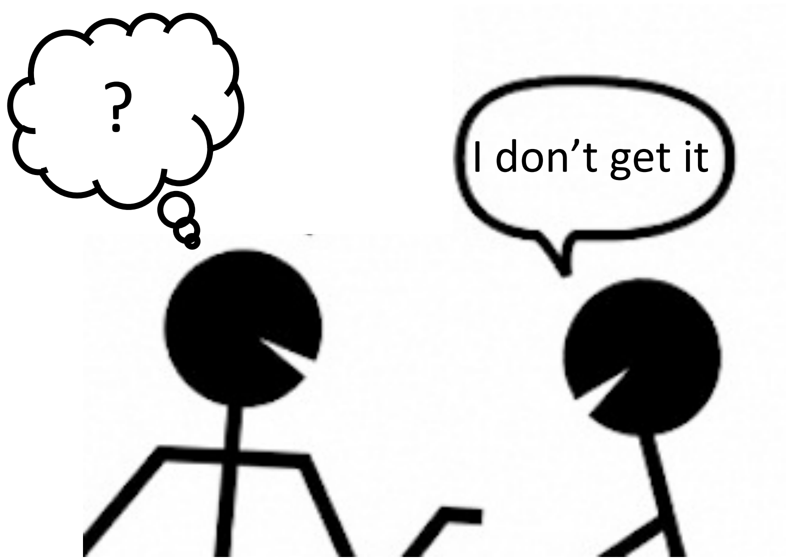
Have You Ever Been …

- An instructor where a student approached you and said they didn't know what they didn't understand?

- A student who has struggled in a class only to be re-taught things that you already know?

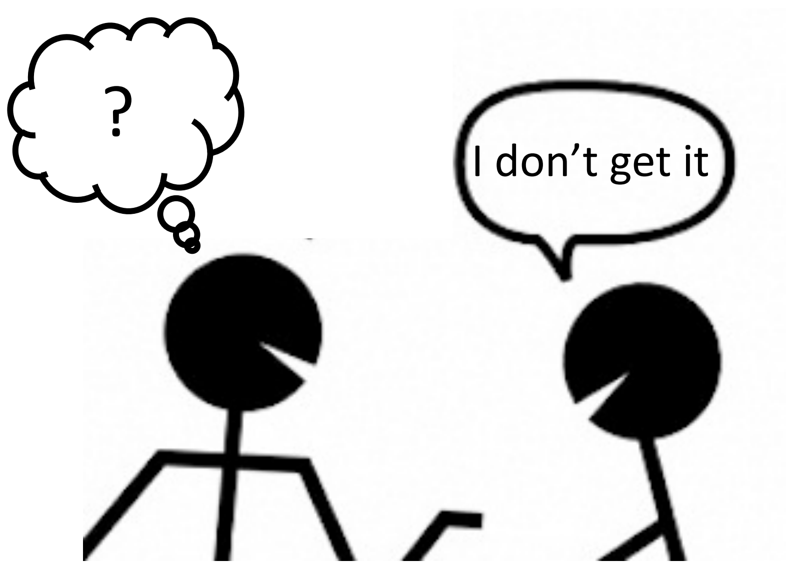
Intelligent Tutoring System (ITS)

- Software that serves as a tutor
- Adapts to student user's needs



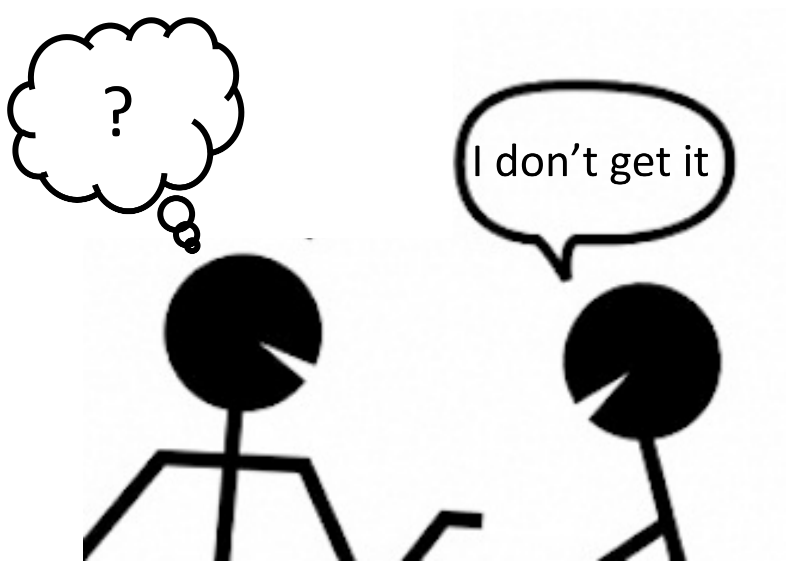
Intelligent Tutoring System (ITS)

- Student’s current level of understanding?
  - Student’s focus, abilities, intentions
  - Student’s past successes, mistakes, learning patterns



Intelligent Tutoring System (ITS)

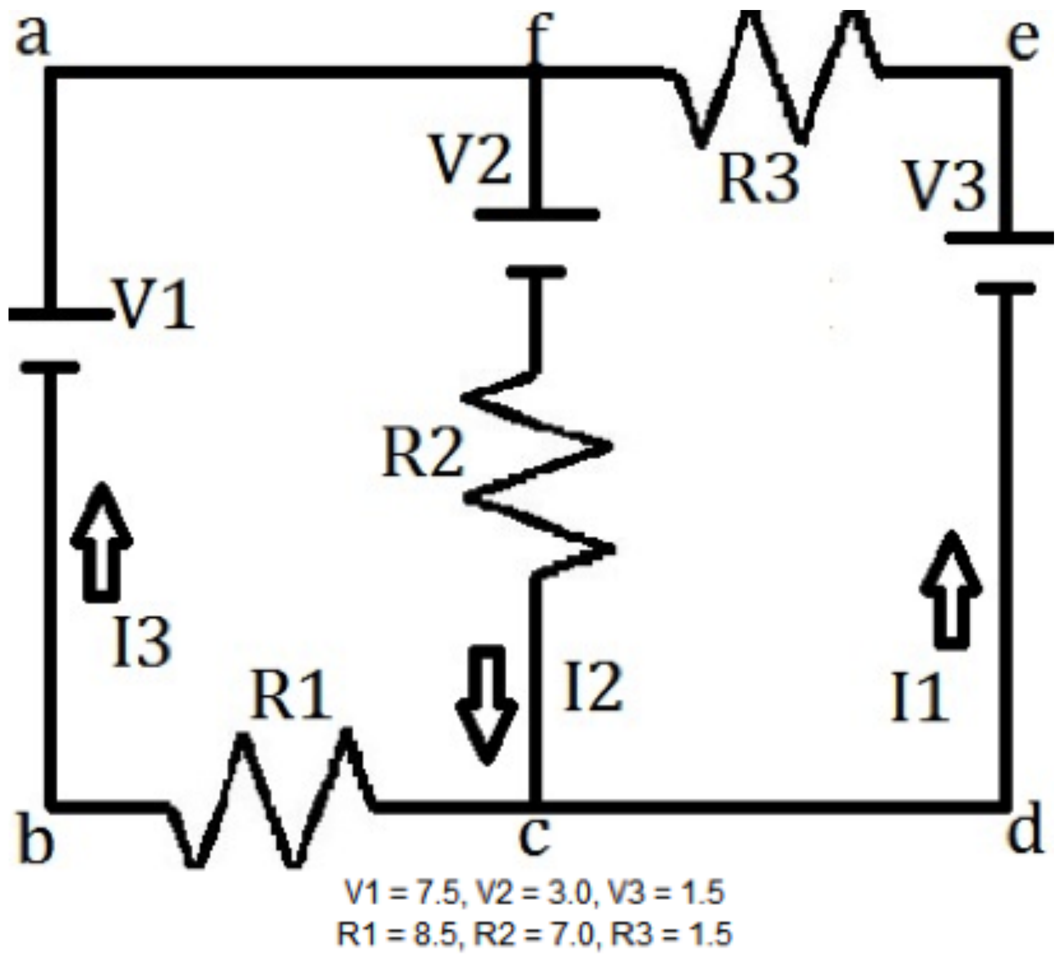
- Student's current level of understanding?
- How best to help student?
  - Watch, show similar example, give hint, give complete solution



Goals of the Project

- Identify a student’s difficulties
- Offer individualized help
- Improve confidence and become comfortable with Physics
- Increase student performance in first year Physics classes
- Increase engagement

Example Problem

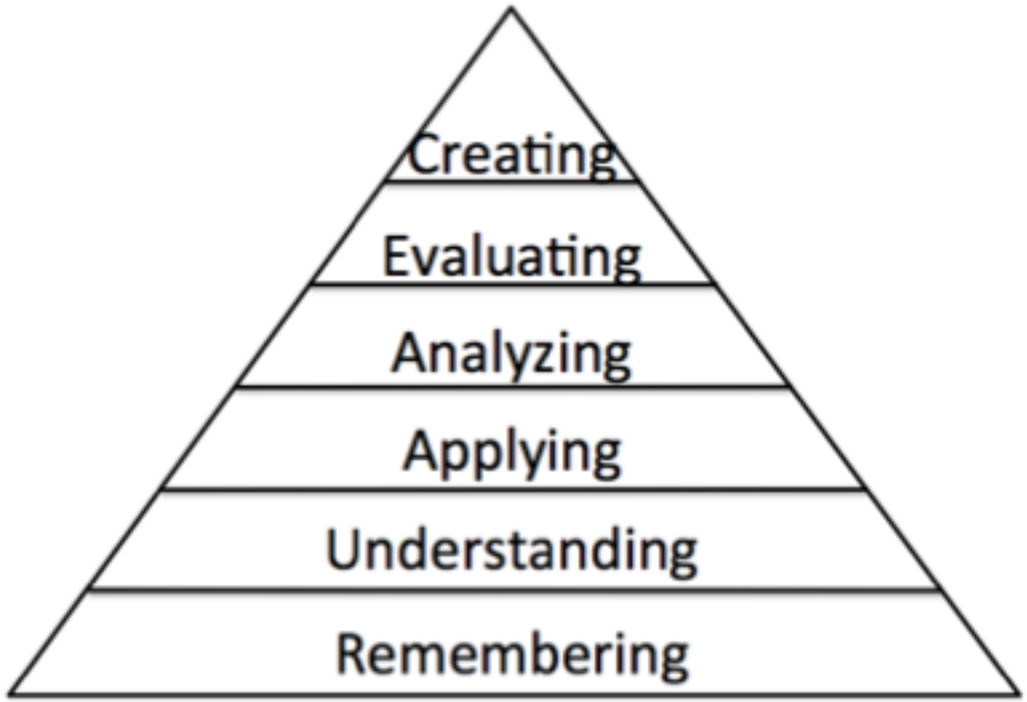


multi-step problem: natural integration for giving feedback

Problem Definition

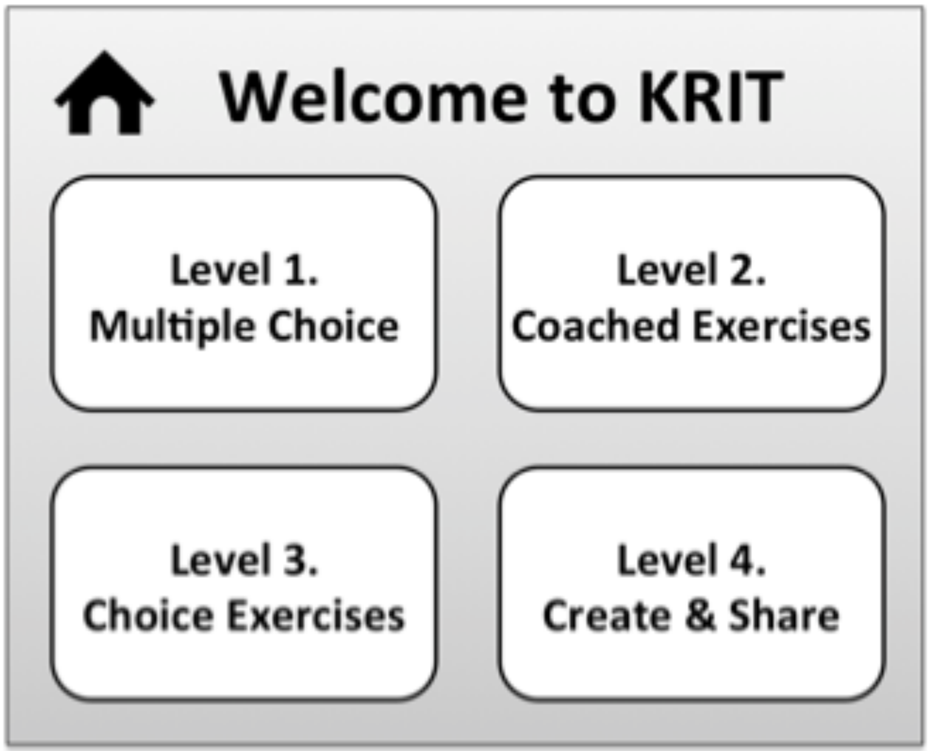
- Circuit complexity:
  - Determined by layout parameters
  - Number of batteries, resistors, junctions
- Objective:
  - Apply Kirchhoff's rules
  - Algebraically solve for one of 3 variables: voltage, resistance, current

Pedagogical Motivation

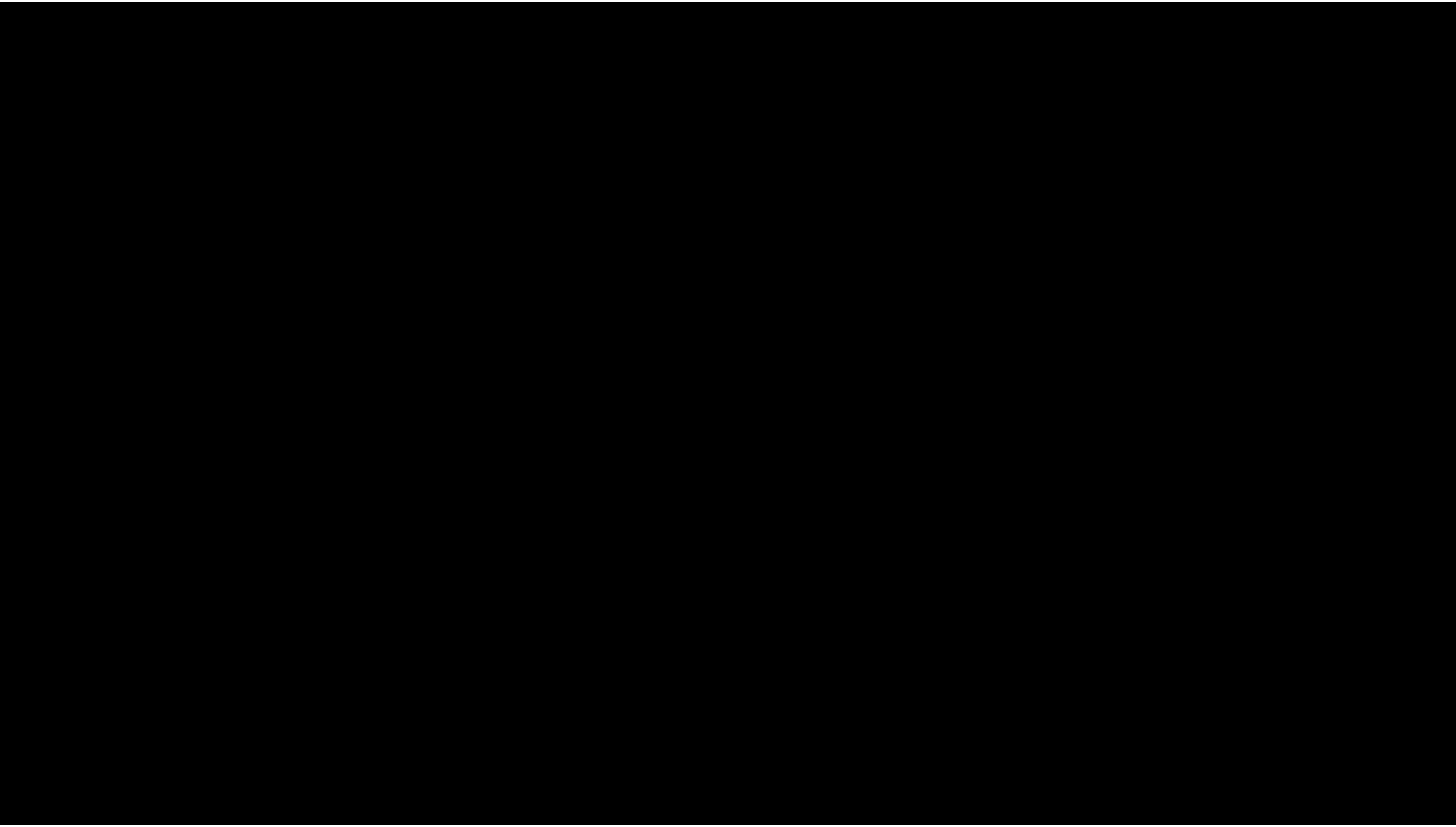


(a) Bloom's taxonomy

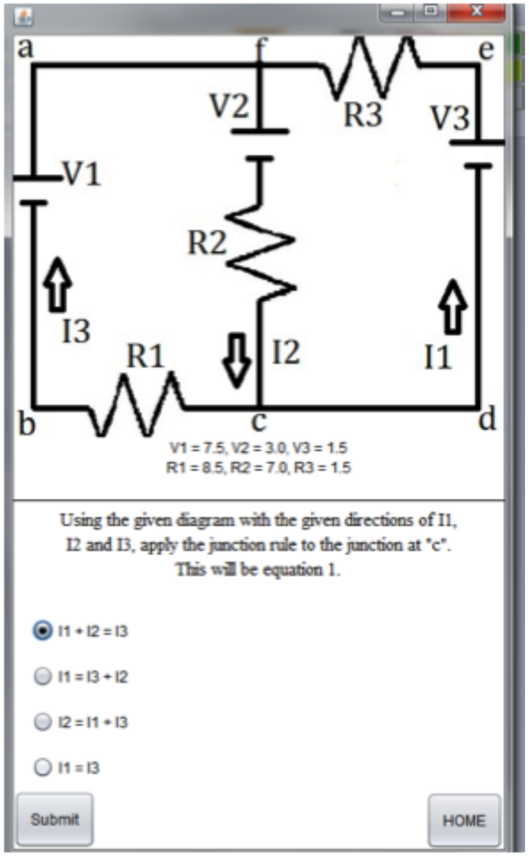
KRIT Difficulty Levels



(b) KRIT Home Screen



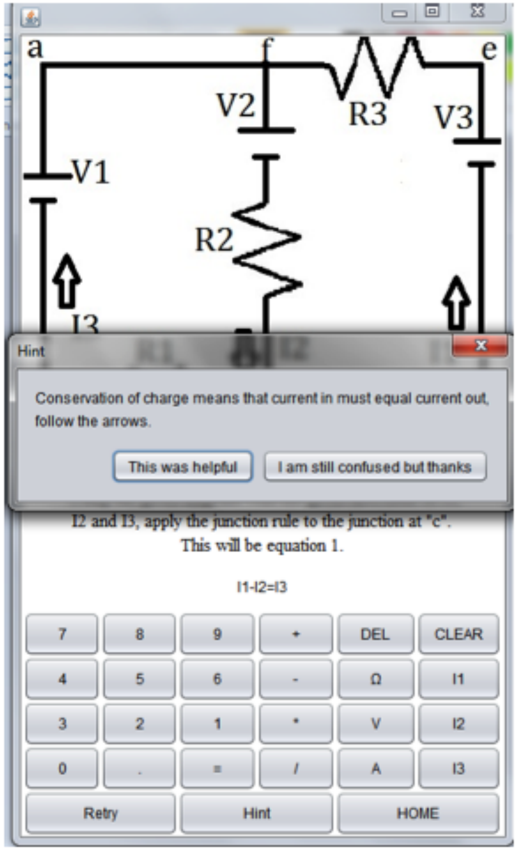
Multiple Choice



(a) Level 1

- multi-step problem
- multiple choice question at each step
- immediate feedback at each step

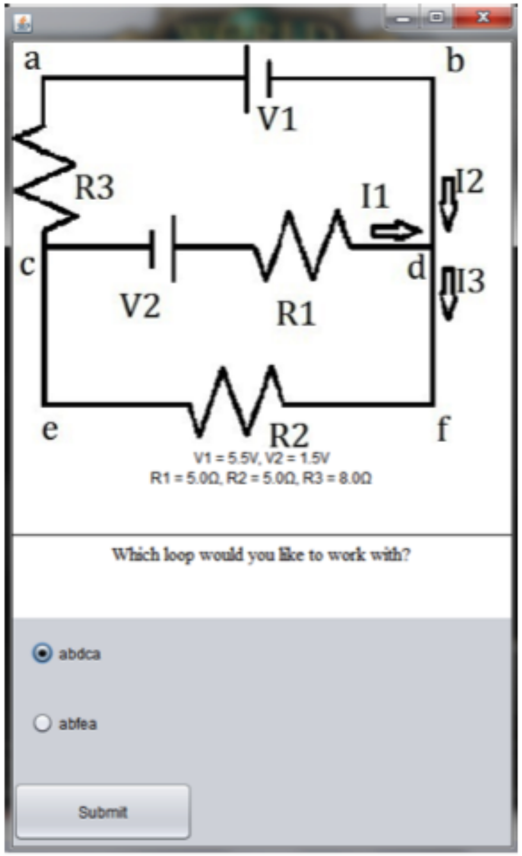
Coached Exercises



(b) Level 2

- input complete answer (e.g., 24Ω)
- probabilistic model to estimate student knowledge areas and independent levels
- hints provided when recommended by model

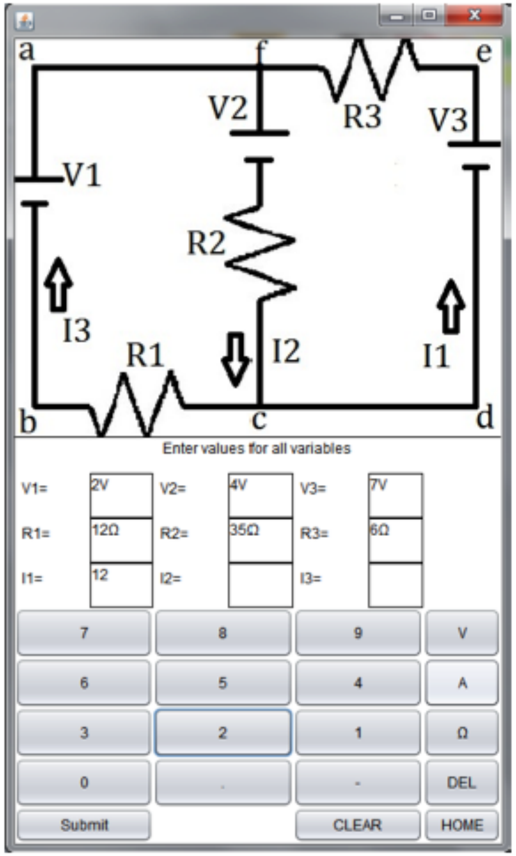
Choice Exercises



(c) Level 3

- modeled after level 2
- added flexibility to choose which order of the steps to solve problem in first
- encourages synthesis of procedural knowledge at a deeper level

Create & Share



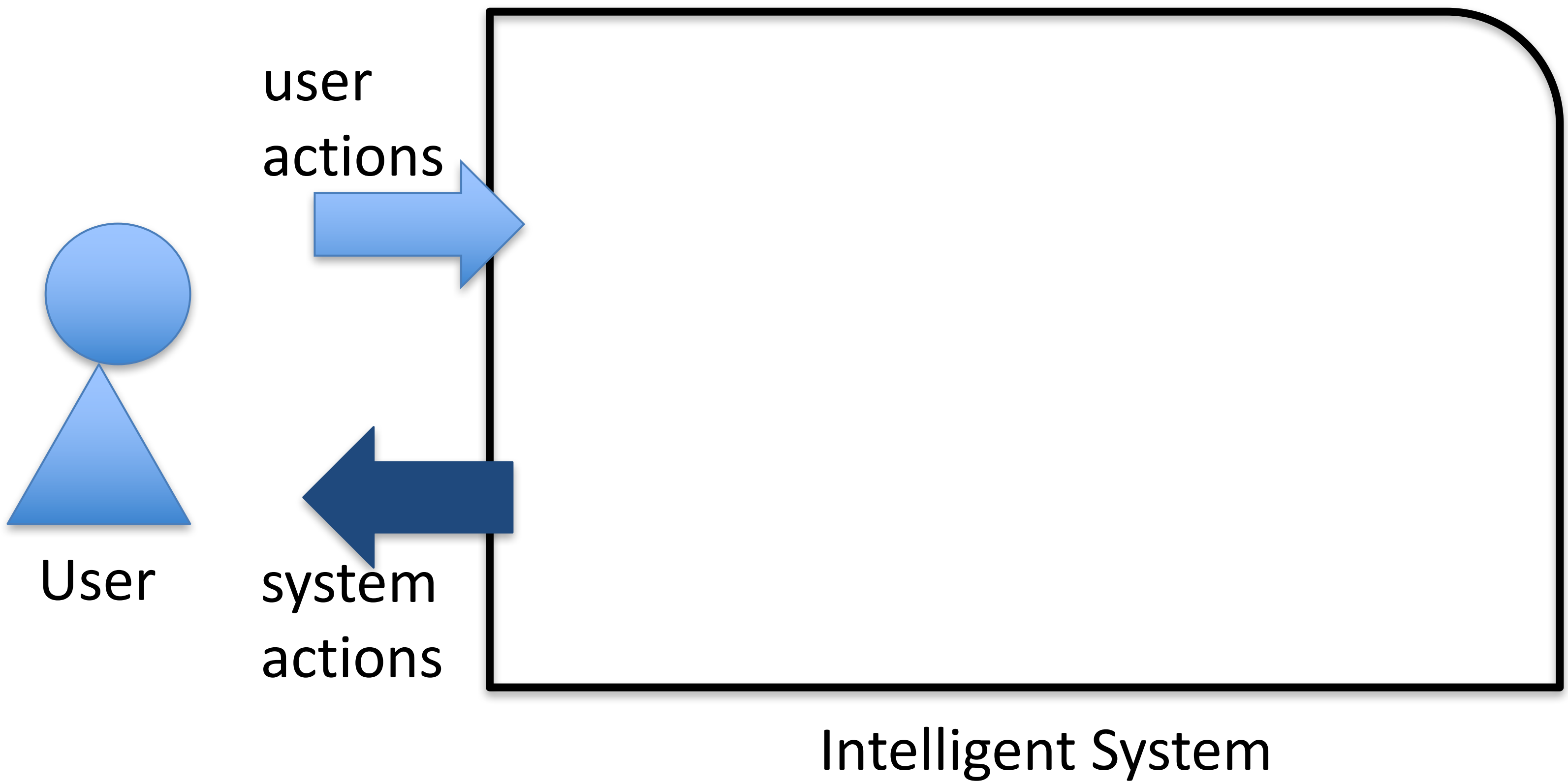
(d) Level 4

- peer learning environment
- total of 4 circuit templates
- custom problems submitted to "Challenge Board"
- student provides new question and answer (ITS verifies answer is correct)

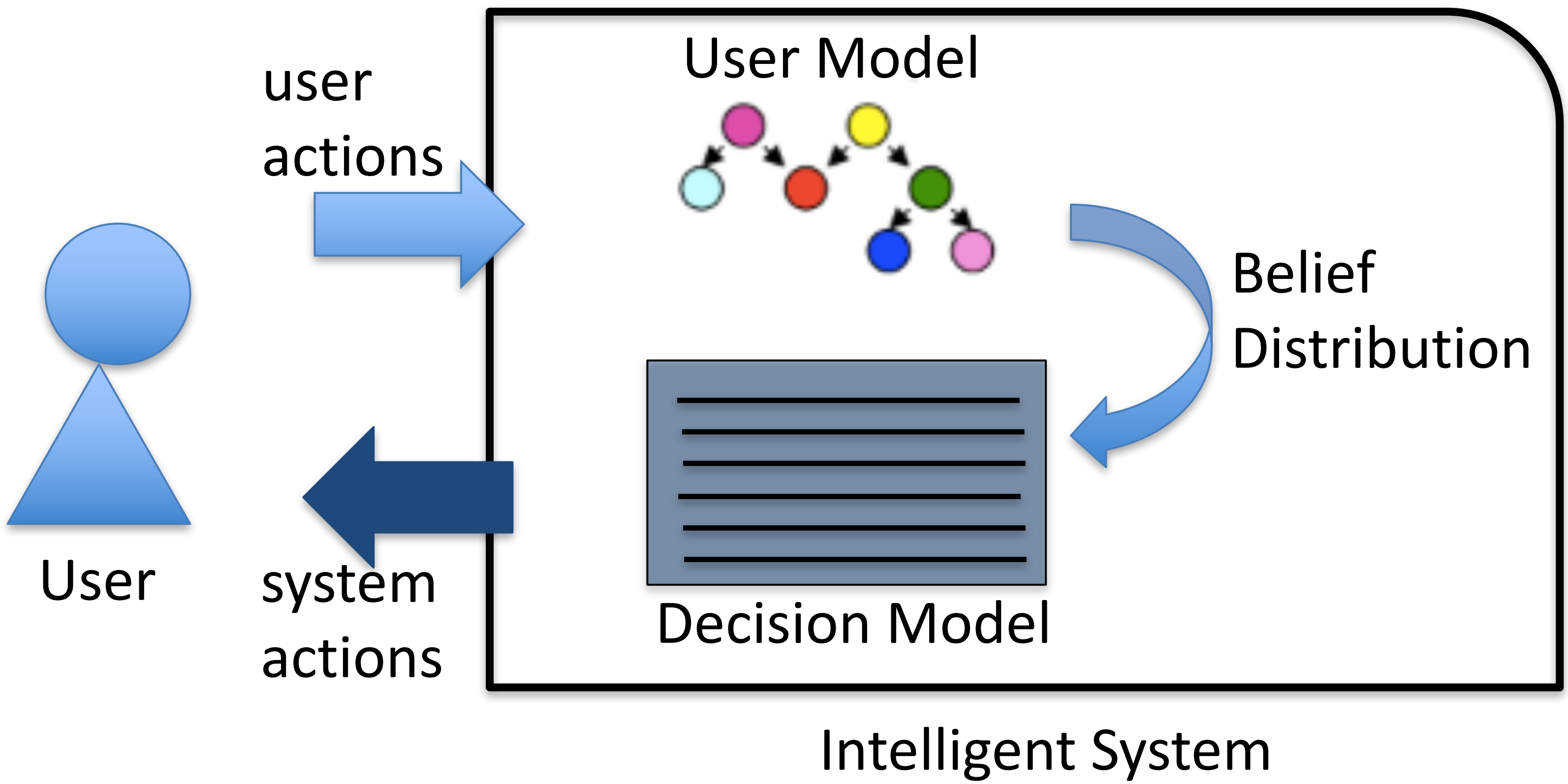
Two Types of Adaptation

- Giving hints:
  - Specific to level 2 (“coached”) exercises
- Changing levels of difficulty:
  - Each level has exercises
  - At the end of an exercise, KRIT will suggest the next exercise (level up or level down)
  - Student can also opt out and select their own

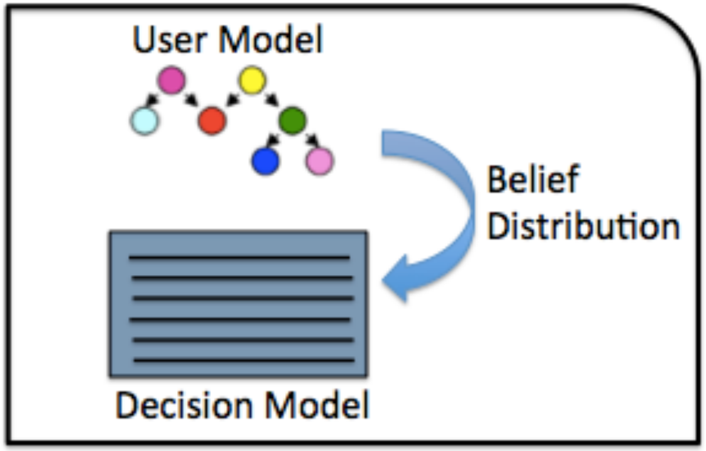
Probabilistic User Modeling and Decision Making



Probabilistic User Modeling and Decision Making



ITS Architecture



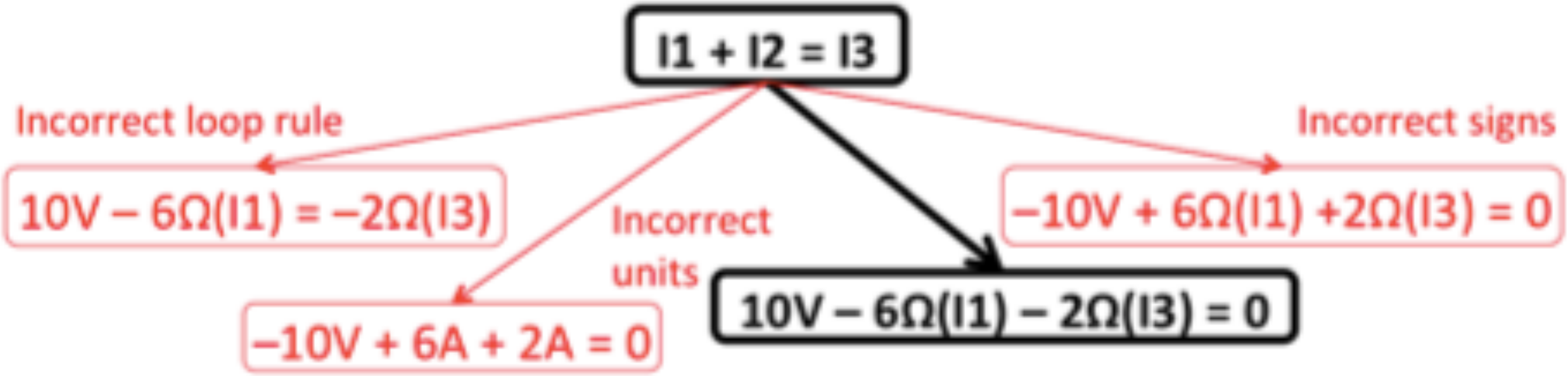
Intelligent System

- User Model:
  - Domain Module
    What does the student know about the domain?
  - Student Module
    What kind of student are we dealing with?
- Decision Model:
  - Tutor Action Selection Module
    What should the tutor do in response?

Domain Module

- Four circuit templates
  - Varies in difficulty (number of free parameters)
  - Automatically generates new exercises on demand
- Each template has corresponding solution graph
  - Outlines steps needed to be completed
  - Outlines typical student errors at each step

Partial Solution Graph



- Black: step-by-step solution
- Red: common misconceptions at that step

Summary of Domain Module

- Identify the structure of the exercises
- Identify the structure of the solutions needed for each type of exercise
- Create a solution graph for each type of exercise
  - Include common mistakes at each step of solution
  - Include hints for each type of mistake

Student Module

- What does the student know about the domain?

- How much help does the student need now?

Student Module

- What does the student know about the domain?
  - Physics knowledge (Kirchhoff's rules)
  - Algebra and units
- How much help does the student need now?

Student Module

- What does the student know about the domain?
  - Physics knowledge (Kirchhoff's rules)
  - Algebra and units
- How much help does the student need now?
  - May prefer to learn on their own instead
  - May need more time to internalize material
  - May be a simple slip and don't need help

Must estimate this information!

What Can We Observe?

- User’s current and past actions
- Algebra?
- Physics?
- Need help?

What Can We Observe?

- User's current and past actions
- Algebra?
  - Numeric accuracy
- Physics?
- Need help?

What Can We Observe?

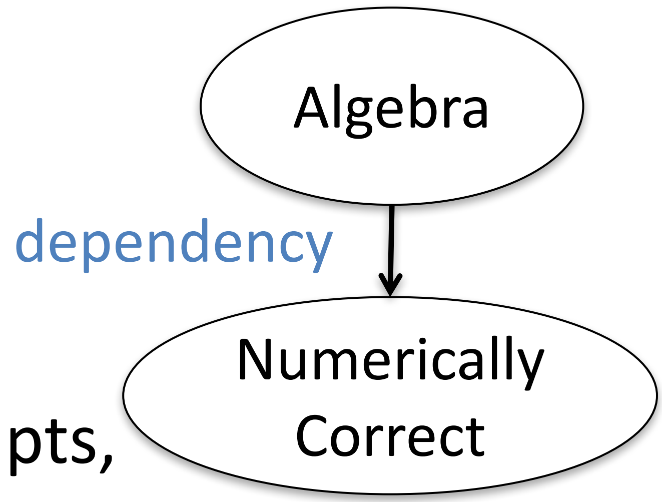
- User's current and past actions
- Algebra?
- Physics?
  - Junction rule: currents balanced
  - Loop rule: all voltages zero, sign correct
  - Measurements: units correct
- Need help?

What Can We Observe?

- User’s current and past actions
- Algebra?
- Physics?
- Need help?
  - Pause, undos, submit blanks, browse around
  - Receptiveness to help: ask hint, read hint, read answer

Example Relationship

- Algebra Knowledge (A)
  - High, medium, low
- Numerically Correct (NC)
  - 70+% correct in all past attempts, 30-70% correct in all past attempts, < 30% correct in all past attempts



Pr(A|NC)

How well you know algebra depends on how many instances you've been correct in the past

Example Relationship (cont.)

- Pr(A|NC) expressed as a conditional probability table (CPT)

Algebra (A)

Numerically Correct (NC)

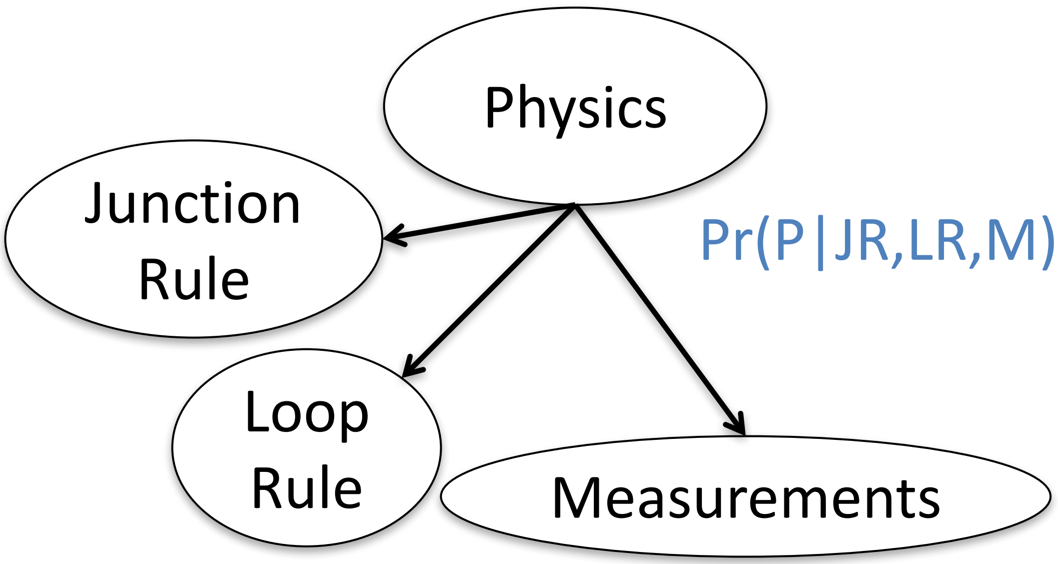
	High	**Medium**	**Low**
70+%	0.85	0.10	0.05
30-70%	0.25	0.60	0.15
< 30%	0.10	0.20	0.70

each row adds up to 1.0

- Pr(A=High|NC=70+%) is 0.85
- Pr(A=Low|NC=70+%) is 0.05

Example Relationship 2

- Physics (P)
  - High, medium, low
- Junction Rule (JR)
  - High, medium, low
- Loop Rule (LR)
  - High, medium, low
- Measurements (M)
  - High, medium, low



How well you know physics depends on how well you know the junction rule, the loop rule, and measurements

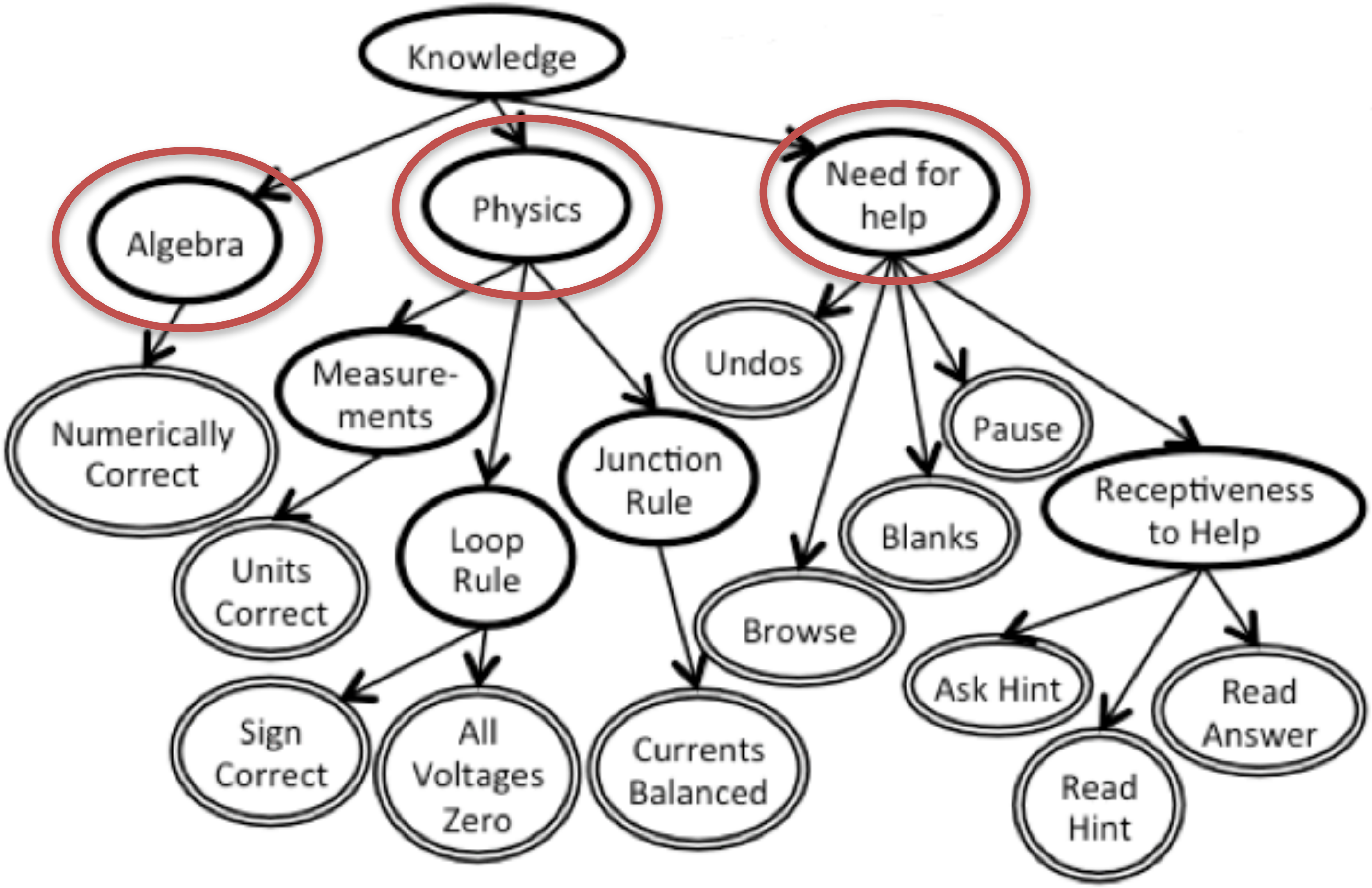
Example Relationship (cont.)

- Pr(P|JR,LR,M) expressed as a CPT

			Physics (P)		
JR	LR	M	**High**	**Medium**	**Low**
High	**High**	**High**	0.95	0.04	0.01
High	**High**	**Medium**	0.85	0.10	0.05
High	**High**	**Low**	0.80	0.15	0.05
High	**Medium**	**High**	0.85	0.10	0.05
High	**Medium**	**Medium**	0.75	0.15	0.10
High	**Medium**	**Low**	0.65	0.20	0.15
High	**Low**	**High**	0.70	0.20	0.10
High	**Low**	**Medium**	0.50	0.30	0.20
High	**Low**	**Low**	0.30	0.40	0.30
Medium	**High**	**High**	0.85	0.10	0.05

. . .

Building a Probabilistic Model

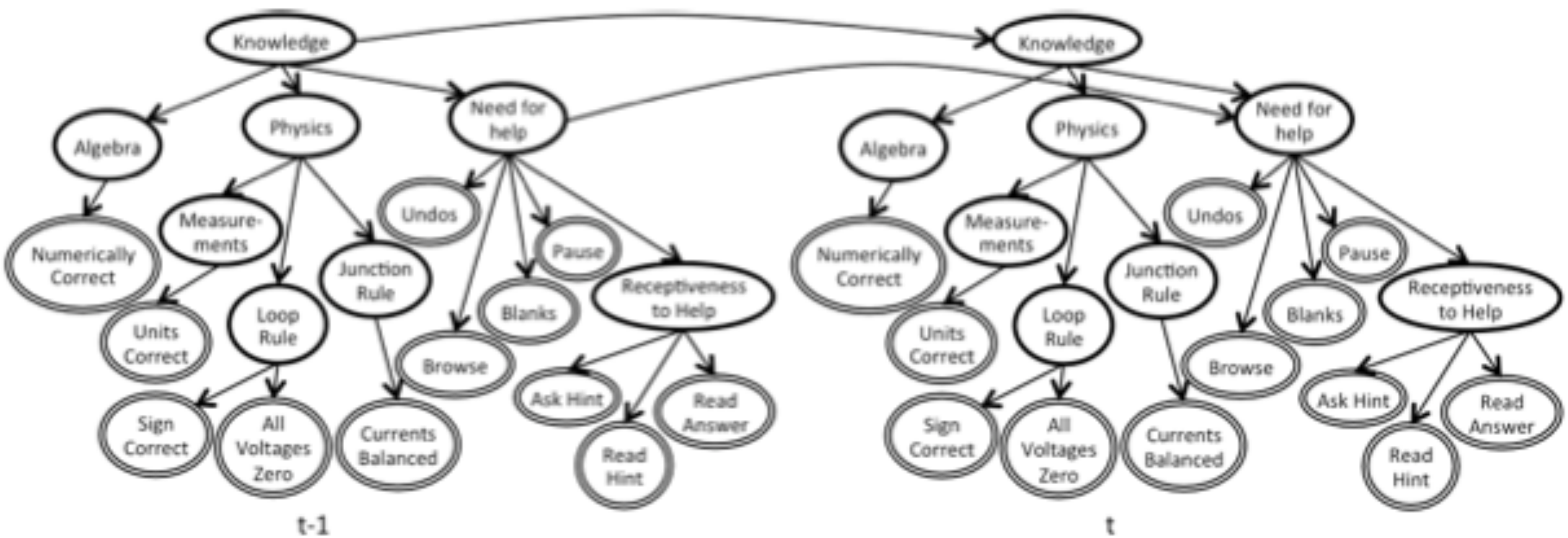


Nodes in single circles are hidden variables

Nodes in double circles are observations

Student Model

Two-slice Dynamic Bayesian Network (DBN)



New CPTs to model temporal relationships:

- $Pr(K_t \mid K_{t-1})$
- $Pr(N_t \mid N_{t-1})$

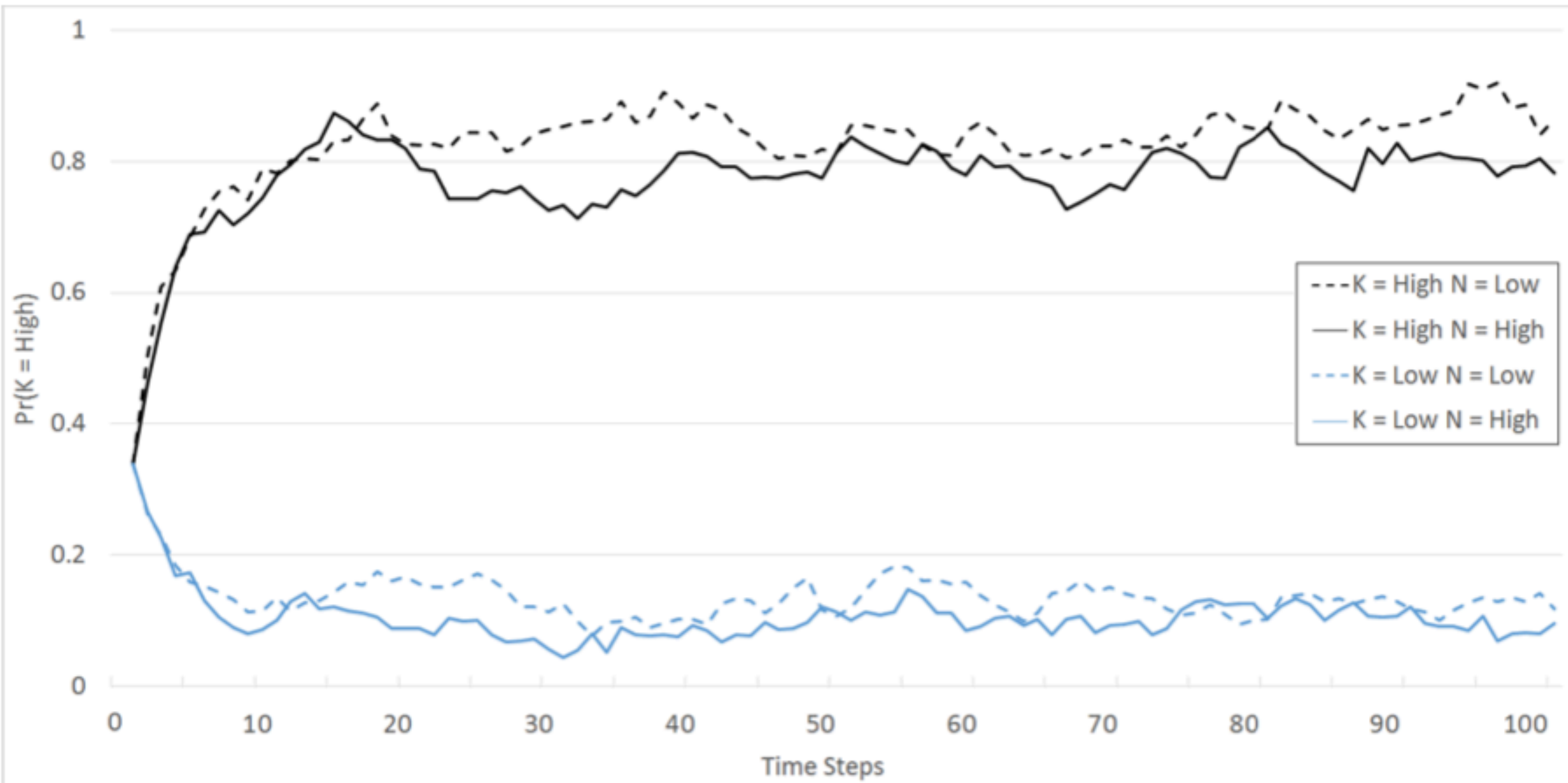
Inference Task

- At each time step:
  - $Pr(K_t, N_t | OBS_1, OBS_2, \ldots, OBS_{t-1})$:
    What is the probability of student's knowledge level and the amount of help needed given *all* the observations (OBS) we've observed in the past?
  - Known as the belief monitoring task
  - Belief distribution over K_t and N_t
- Exact inference computed via clique tree algorithm

Simulation Experiments

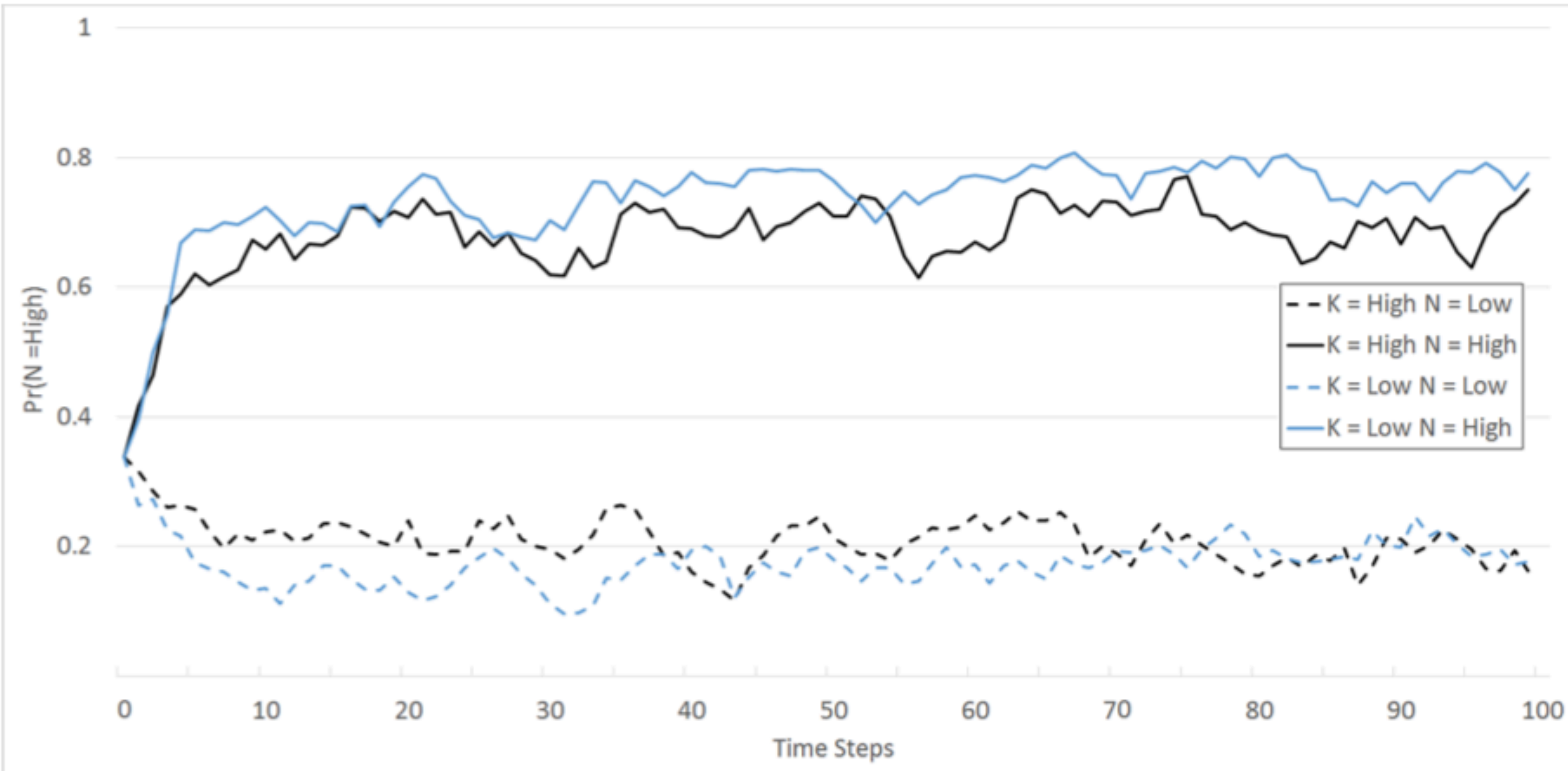
- Created 4 artificial students with fixed values of K and N (high/low combos)
- Using our DBN:
  - Initialize K and N with the fixed student type
  - Repeatedly:
    - Randomly sample DBN to get observations at time t
    - Compute $Pr(K_t, N_t | OBS_{1:t})$
    - Compute action with maximum expected utility A_t
    - Record actual induced cost/reward on student

Inference Results



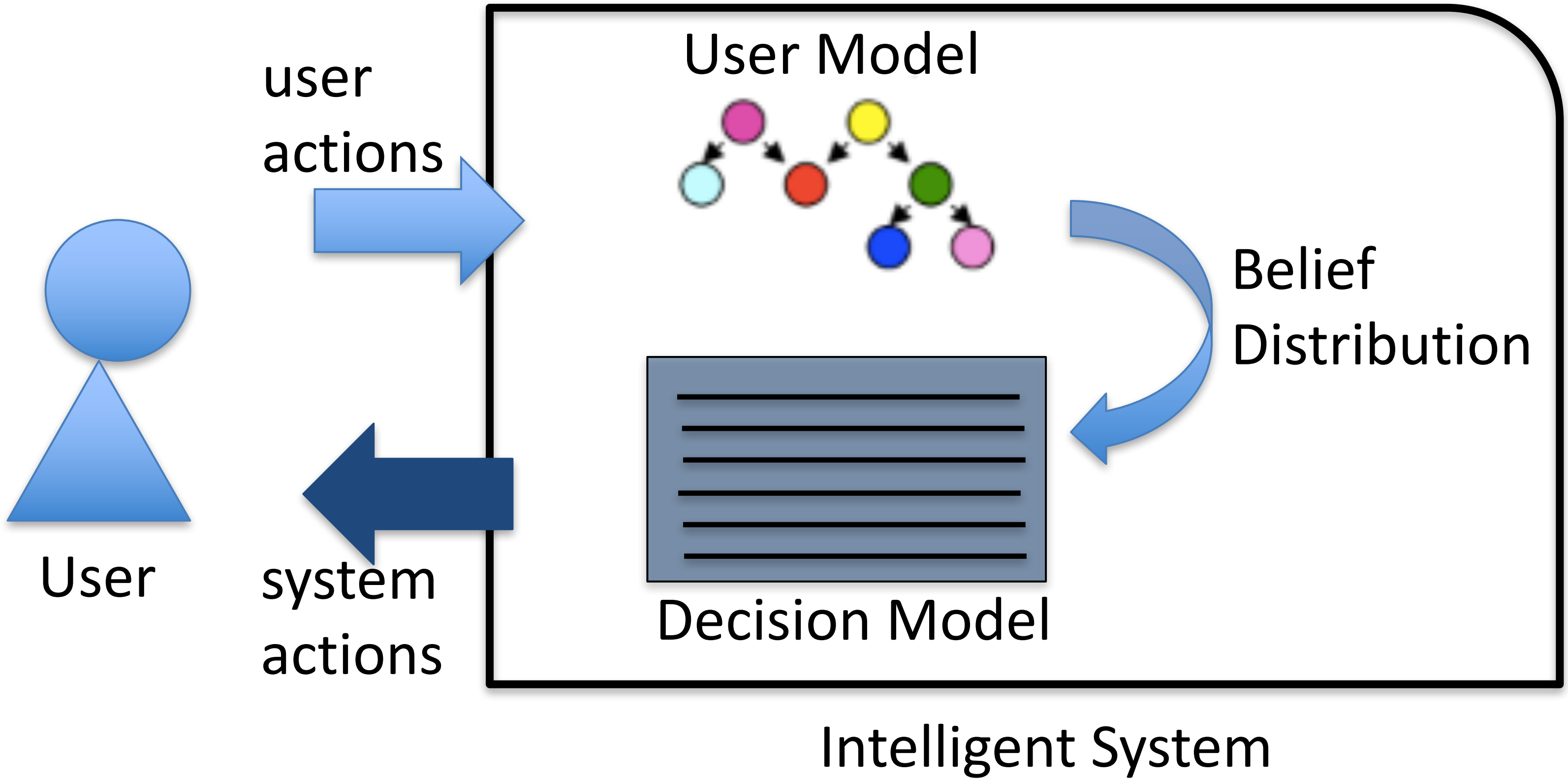
(a) $Pr(K = high)$

Inference Results



(b) $Pr(N = high)$

Probabilistic User Modeling and Decision Making



Once we know what type of user we are working with, what should the system do?

Tutor Action Selection Module

- Possible tutor actions at a given time (A_t):
  - Provide a hint
  - Give an explanation with correct answer
  - Do nothing (let student continue working)

Tutor Action Selection Module

- Possible tutor actions at a given time (A_t):
  - Provide a hint
  - Give an explanation with correct answer
  - Do nothing (let student continue working)
- Define expected utility of each action as:
  - $EU(A_t) = U(A_t | K_t, N_t) \Pr(K_t, N_t)$

How good is an action in expectation of K_t and N_t

Tutor Action Selection Module

- Possible tutor actions at a given time (A_t):
  - Provide a hint
  - Give an explanation with correct answer
  - Do nothing (let student continue working)
- Define expected utility of each action as:
  - $EU(A_t) = \Sigma_{KtNt} U(A_t | K_t, N_t) \Pr(K_t, N_t)$

How good is an action in expectation of K_t and N_t

- Take action with maximum expected utility

Adaptive Tutoring Strategy

- Capture student type using joint probability $Pr(K_t,N_t)$

Adaptive Tutoring Strategy

- Capture student type using joint probability $Pr(K_t, N_t)$
- Capturing individual behaviour and preferences using $U(A_t | K_t, N_t)$

Adaptive Tutoring Strategy

- Capture student type using joint probability $Pr(K_t,N_t)$
- Capturing individual behaviour and preferences using $U(A_t|K_t,N_t)$
  - Giving hints to student who does not need help induces a cost, and that cost increases if the student's knowledge is high

    Don't interrupt student
  - Giving full explanation to student who does not understand material and needs a lot of help results in a high reward

    Help user get unstuck

Simulation Results with Adaptive Actions

- Same simulation setup as earlier
- Average utility collected at each time step after carrying out an action

$K = high$	
N	**Avg. Utility**
low	-22.14
medium	-15.24
high	-6.62

$K = medium$	
N	**Avg. Utility**
low	-8.20
medium	6.60
high	17.80

$K = low$	
N	**Avg. Utility**
low	9.76
medium	24.24
high	37.20

- Most beneficial for students with low knowledge and high neediness

Adaptive Actions

- Within an exercise:
  - Possible tutor actions such as:
    - Provide a hint
    - Give an explanation with correct answer
    - Do nothing (let student continue working)
  - Depending on the user type
- Across exercises:
  - Go to an easier exercise
  - Go to a harder exercise
  - Stay at the same level of exercises

Choosing the Next Exercise

- Simple heuristic:
  - If $\Pr(K_t) > \tau_1$
    choose an exercise at a higher level of difficulty

Give more challenge

Choosing the Next Exercise

- Simple heuristic:
  - If $Pr(K_t) > \tau_1$ choose an exercise at a higher level of difficulty — Give more challenge
  - If $Pr(K_t) < \tau_2$ choose an exercise at a lower level of difficulty — Don’t frustrate student

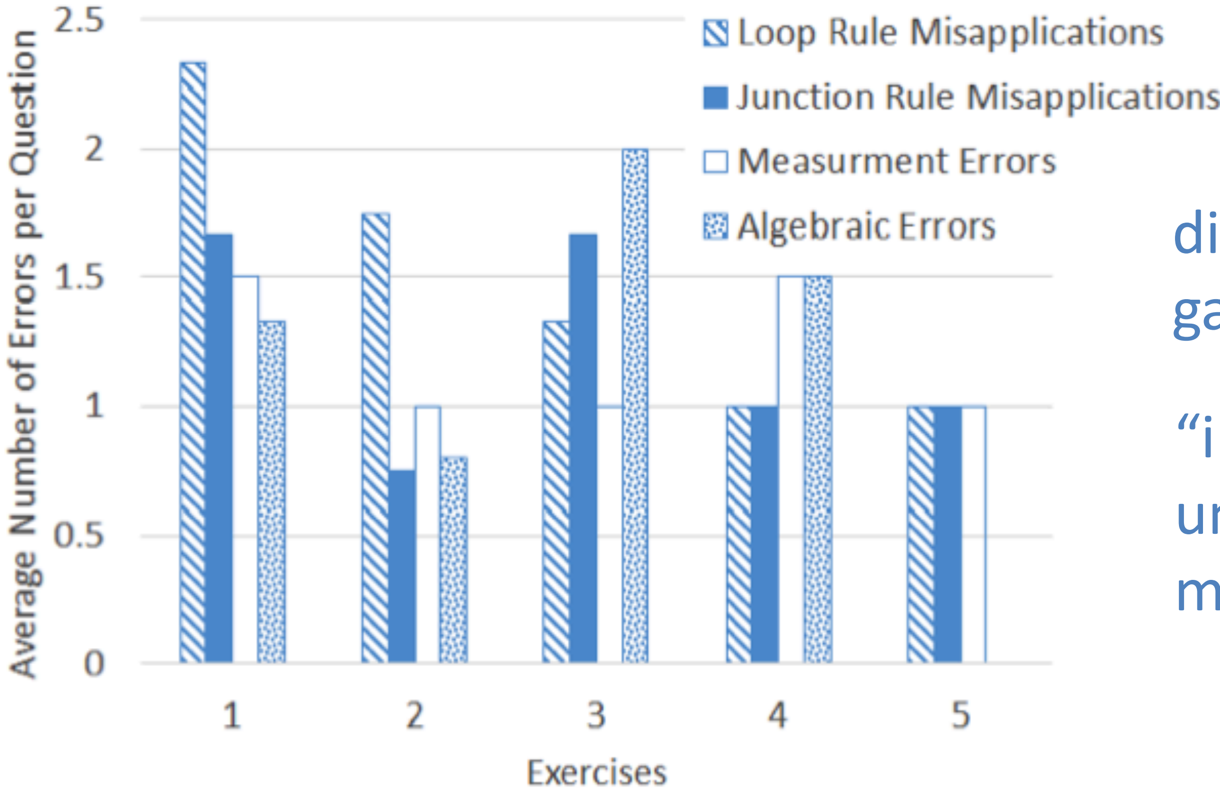
Choosing the Next Exercise

- Simple heuristic:
  - If $Pr(K_t) > \tau_1$ choose an exercise at a higher level of difficulty — Give more challenge
  - If $Pr(K_t) < \tau_2$ choose an exercise at a lower level of difficulty — Don't frustrate student
  - Else choose an exercise at the same difficulty level — Keep working at it
- Thresholds currently set to $\tau_1 = 0.6$ and $\tau_2 = 0.4$

Pilot User Study

- Six first year Physics students
  – Questionnaire on attitudes toward Physics (adapted from Intrinsic Motivation Inventory)
  – Pre-test on Kirchhoff's rules
  – Knowledge pre-test autograded to compute prior for $Pr(K_0)$ in the DBN
  – Use KRIT for 45 min, and optionally 30 min more
  – Post-test on Kirchhoff's rules, questionnaire on attitudes, questionnaire on usability

Study Results

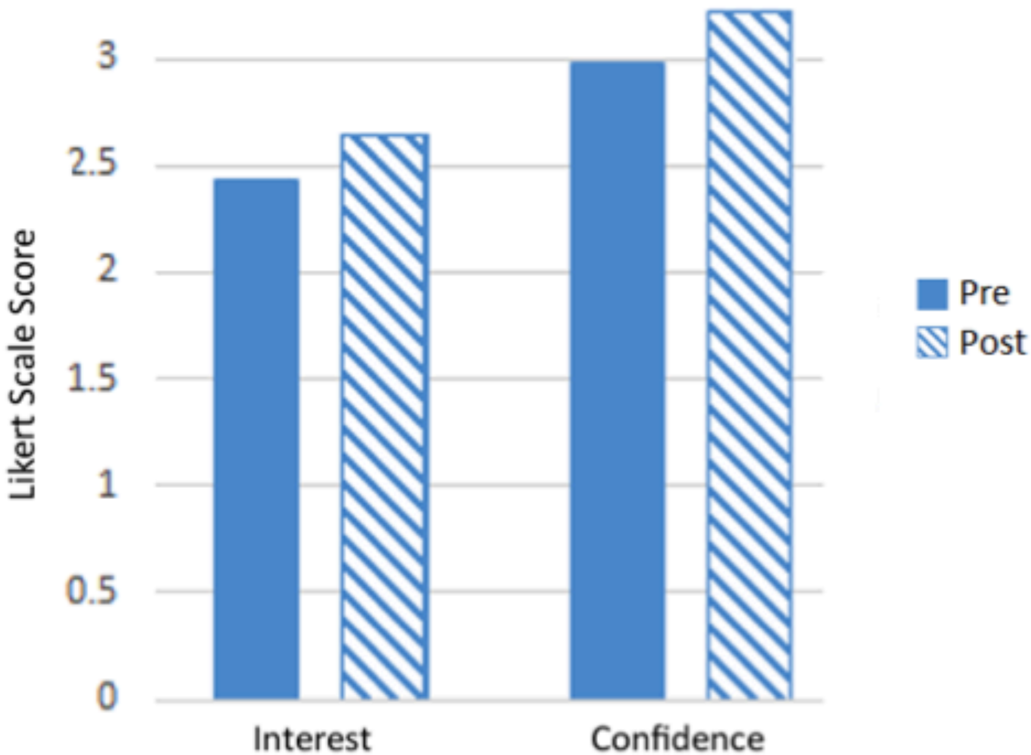


(a) Knowledge

did not expect learning gains in 45 min

“increased their understanding of the material”

Study Results



(b) Attitudes

"more engaging than classroom exercises or homework"

Summary

- Personalized tutoring software for Kirchhoff's rules
- Exercises at 4 levels of difficulty
- Built a probabilistic student model to infer knowledge level and need for help
- Adaptively provide hints, full explanation, or do nothing
- Simulation results showing theoretical value
- Encourage user feedback from pilot study