Toward Experiential Utility Elicitation for Interface Customization

Bowen Hui and Craig Boutilier

Department of Computer Science
University of Toronto

Uncertainty in Artificial Intelligence (UAI'08)

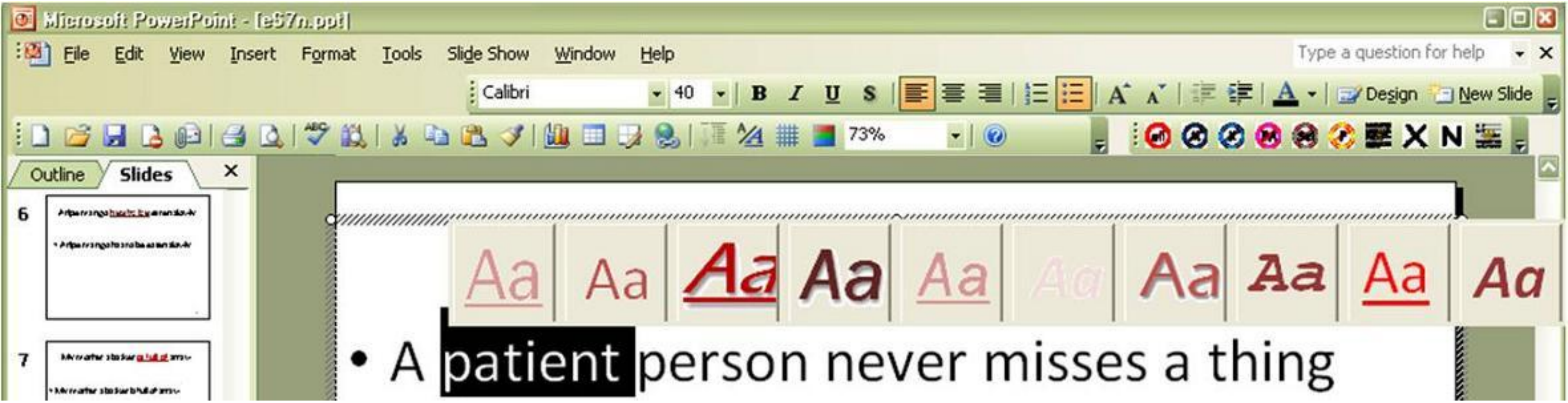
Need for Software Customization

- Problems from industry practices:
  - Varying user needs and preferences
  - One-size-fits-all solution
  - Lost in interface/functionality
- Most affected users
  - People with cognitive, sensory, motor impairments
  - Elderly people
  - Children
  - Novices

What is Software Customization?

- Customization dimensions:
  - Functionality
  - Navigation
  - Presentation
- Focus: customize nature of **help**
  - Intelligent assistance domain
  - Adaptable vs. adaptive vs. mixed-initiative
  - Encompassing decision-theoretic framework

Customizing Assistance



- Highlighting example illustrates:
  - A way to minimize user effort
  - Opportunities created by repetitive goals
  - Customization in presentation dimension
  - Customization in functionality dimension (what,when)

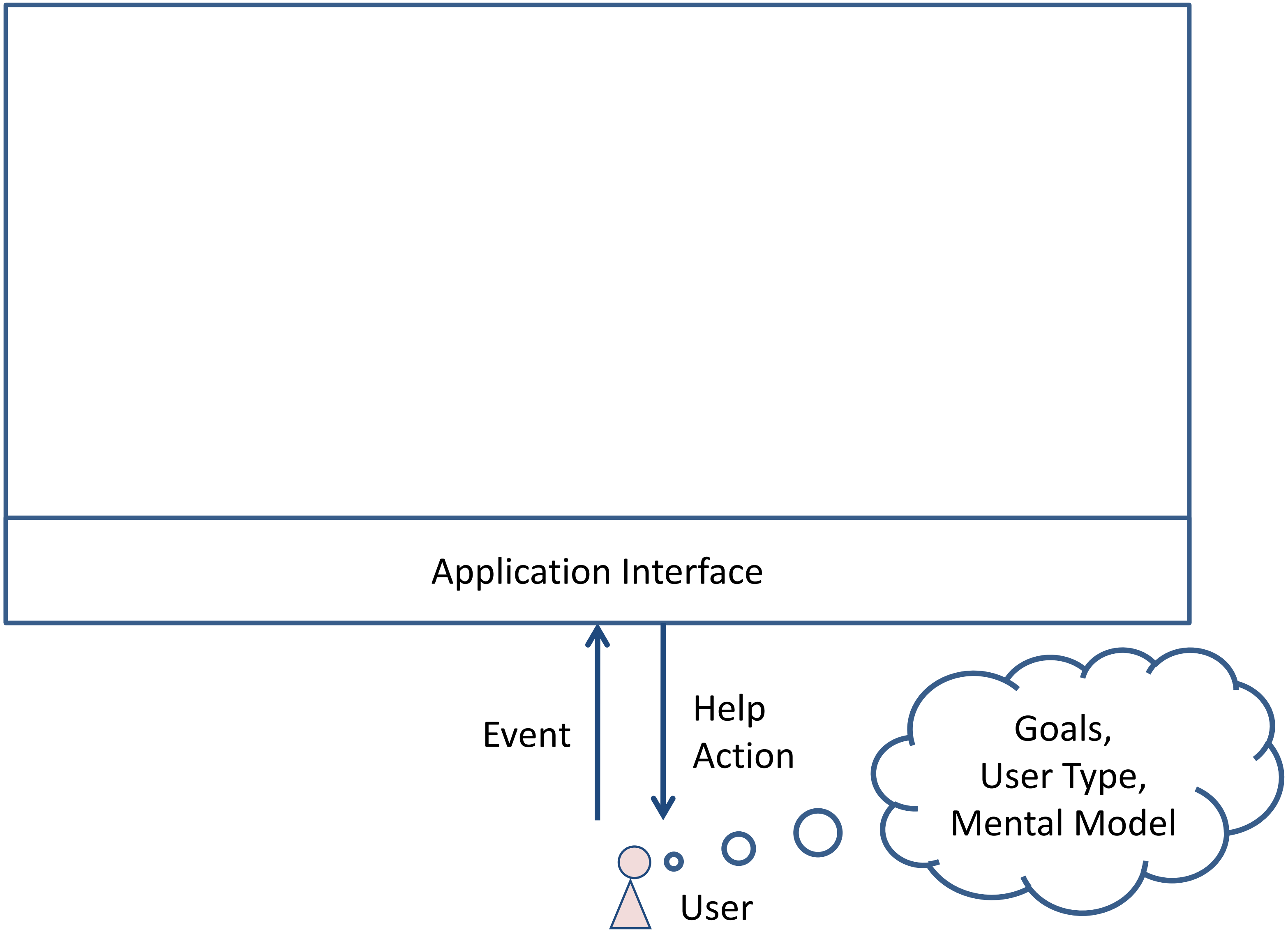
Preferences for Assistance

- What, when, where, how, why?
- Focus on toolbar suggestions for highlighting
  - Quality of suggestion
  - Number of suggestions
  - Whether user needs help

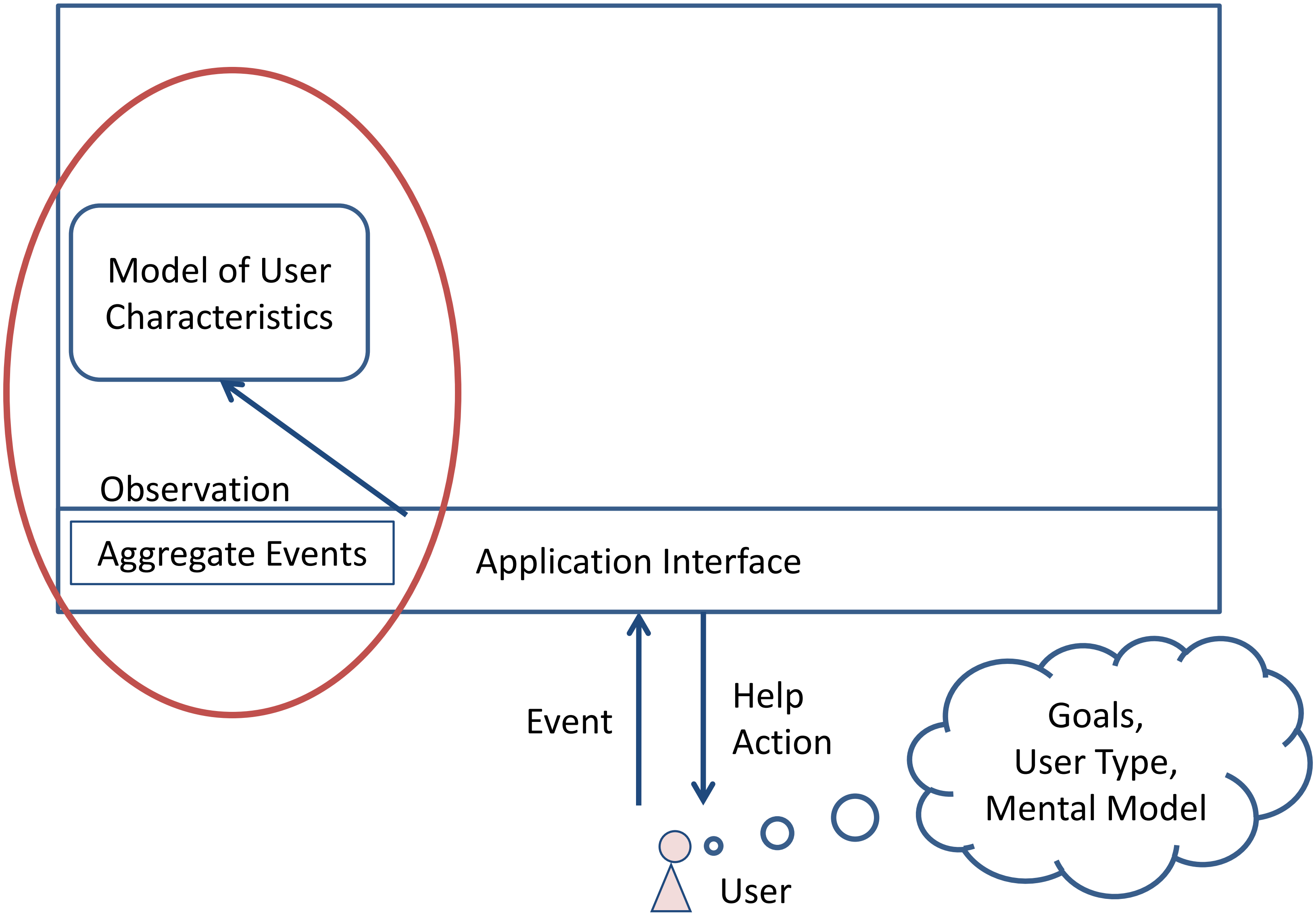
Decision-Theoretic Customization

- Goals: highlighting in PowerPoint
- Assistance: toolbar suggestions
- Partially observable Markov decision process (POMDP)
  - Uncertainty in user goal/features
  - Noise in observation
  - Value and costs of assistance (preferences)
  - Assistance that max. (long-term) expected utility

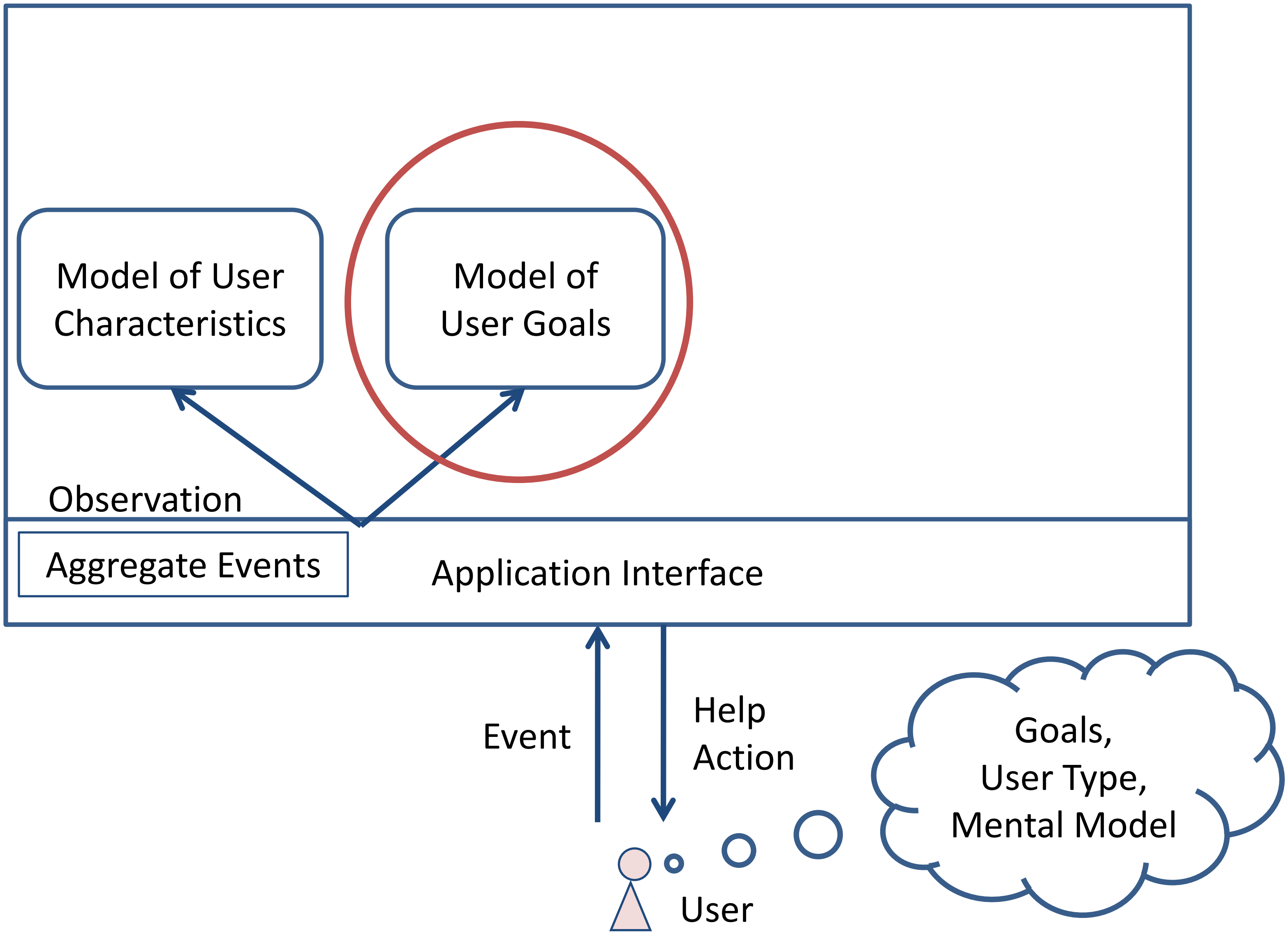
Intelligent Assistance as a Planning Problem



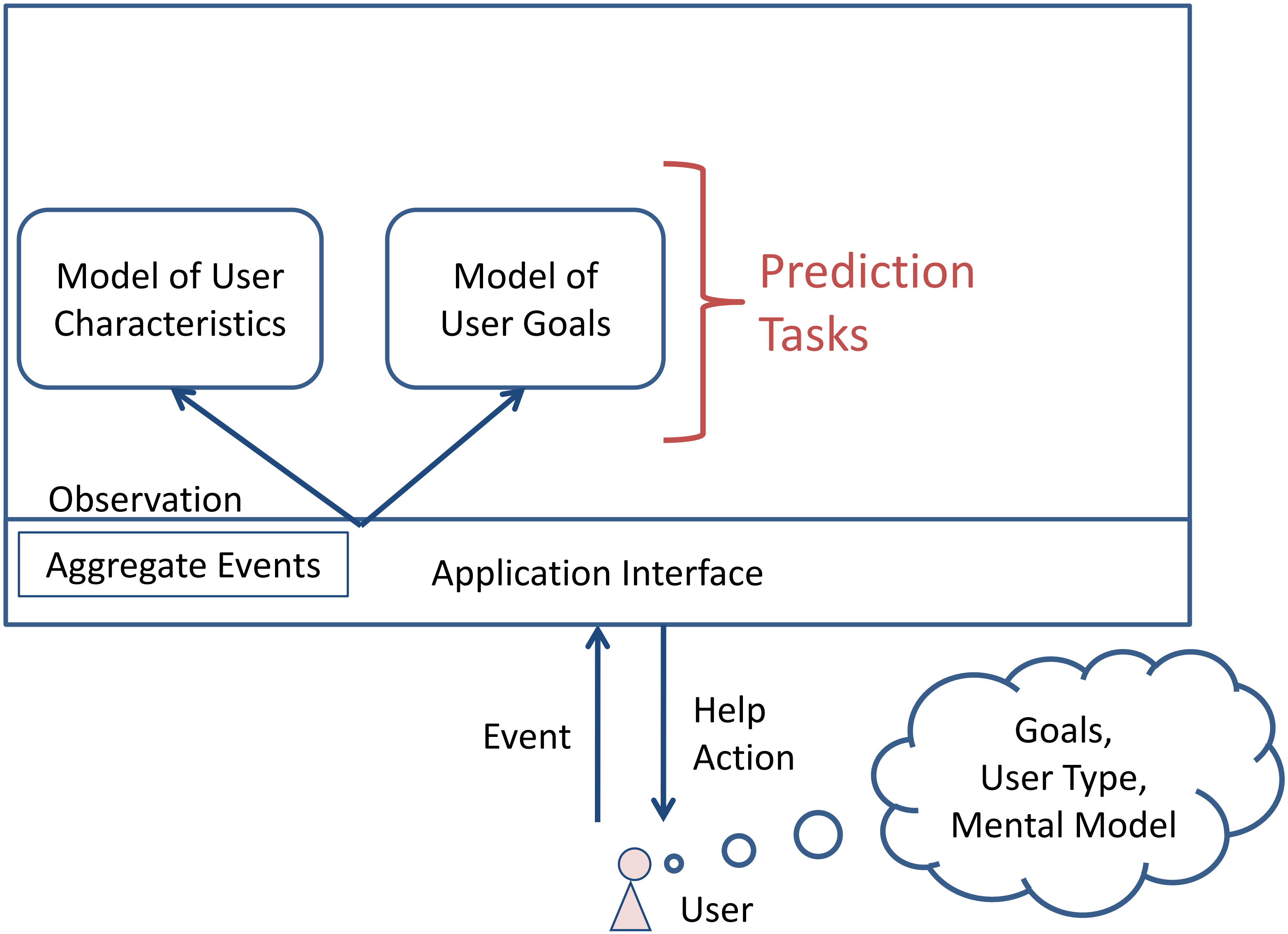
Intelligent Assistance as a Planning Problem



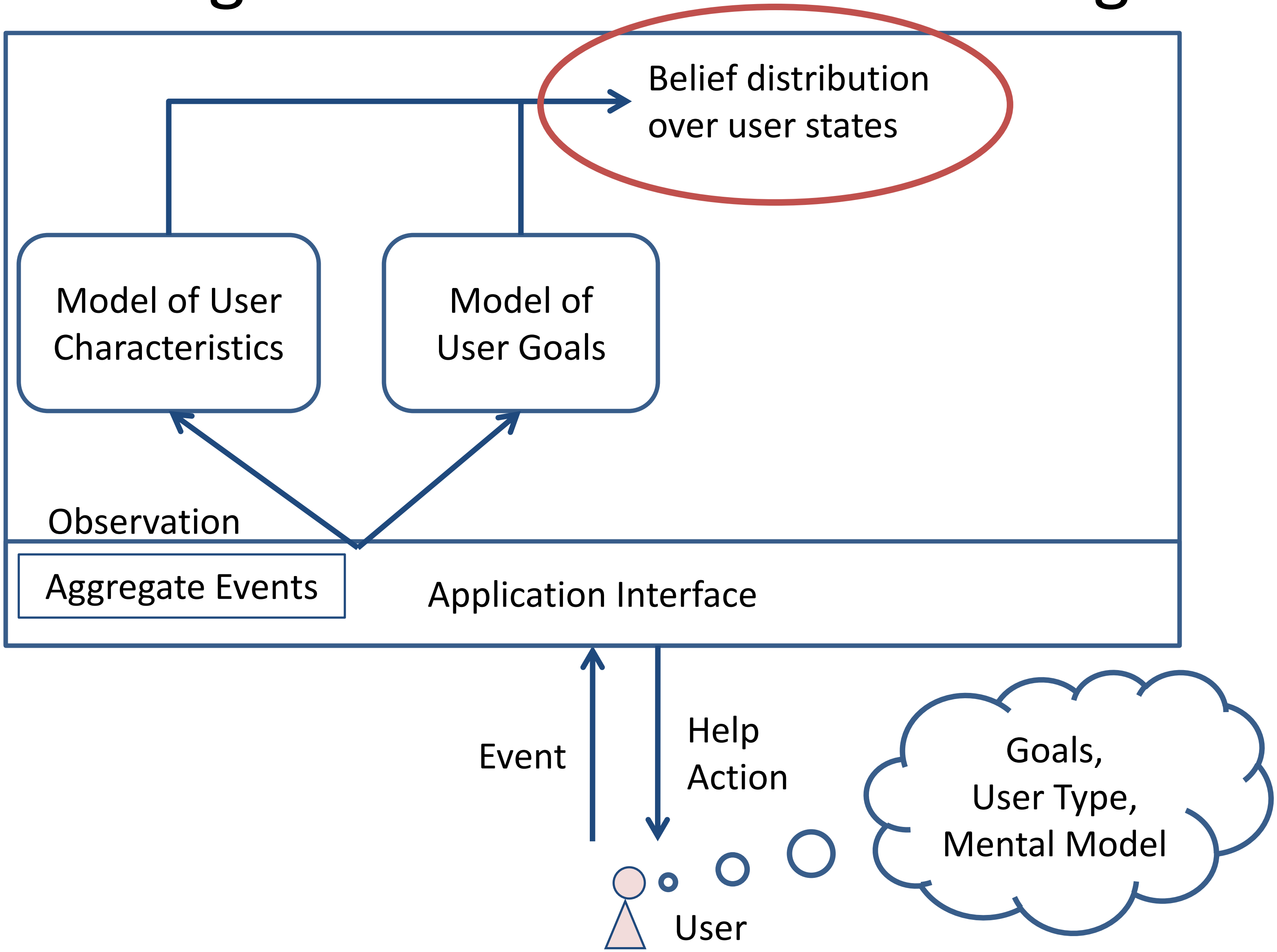
Intelligent Assistance as a Planning Problem



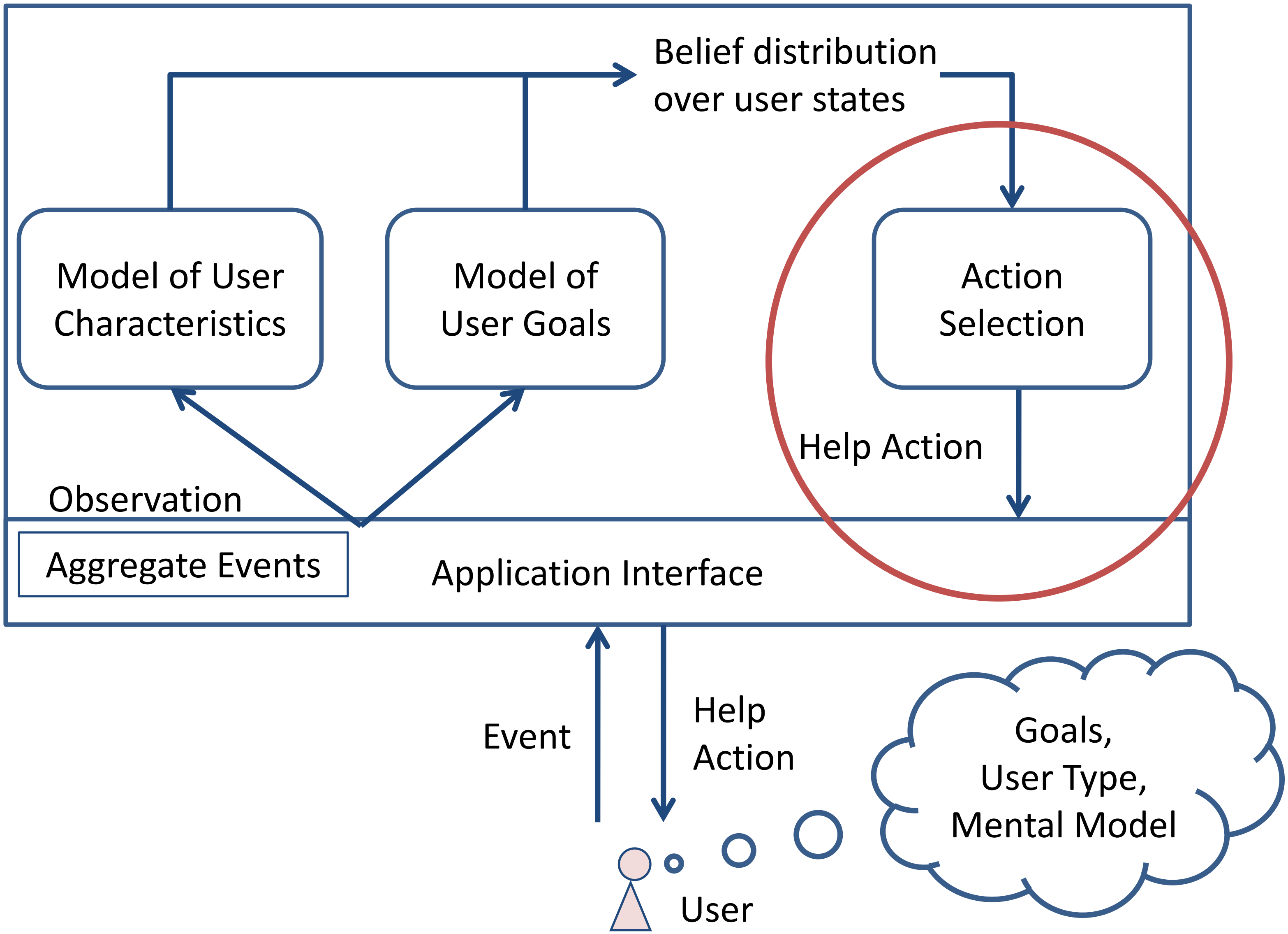
Intelligent Assistance as a Planning Problem



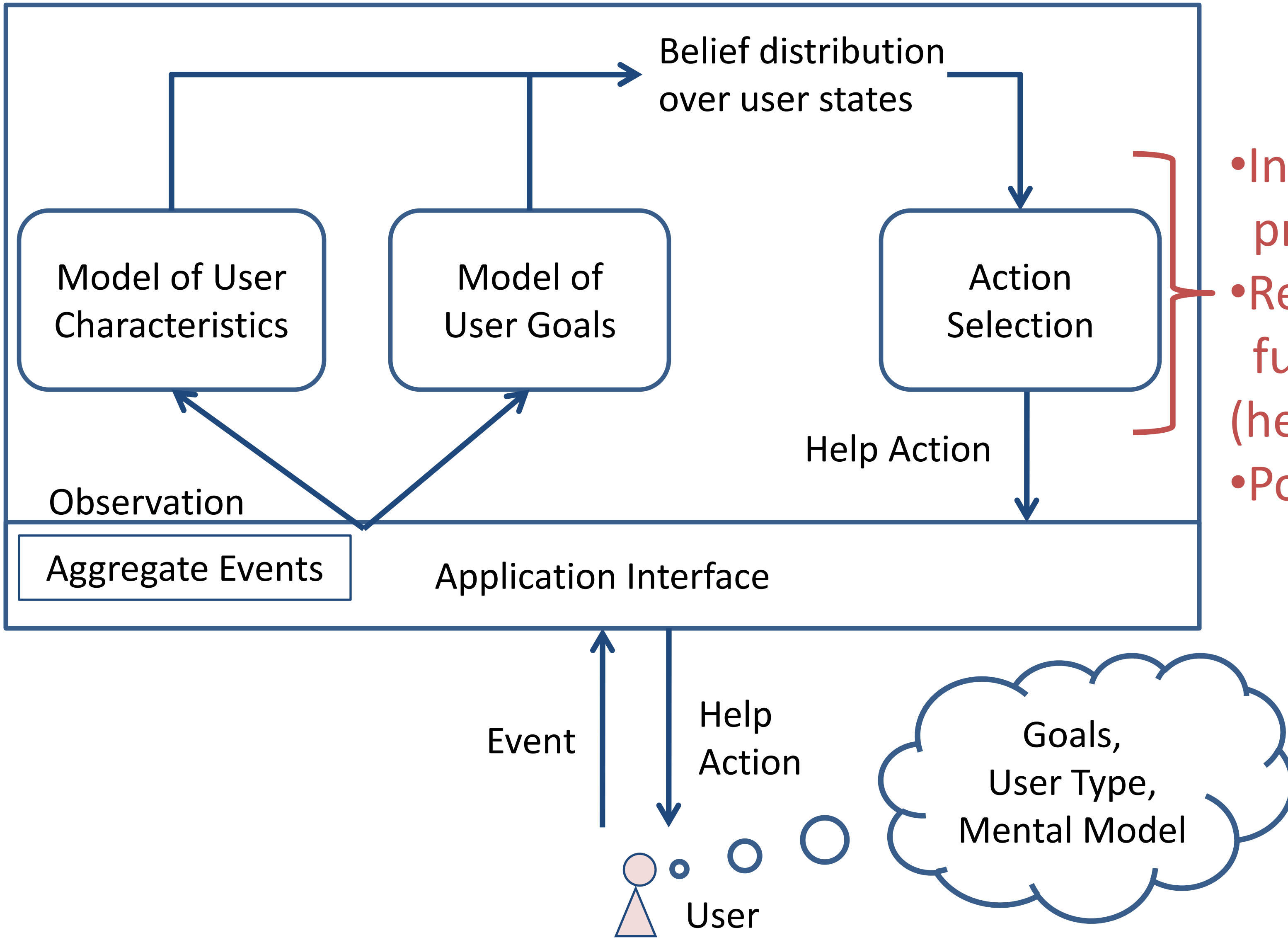
Intelligent Assistance as a Planning Problem



Intelligent Assistance as a Planning Problem



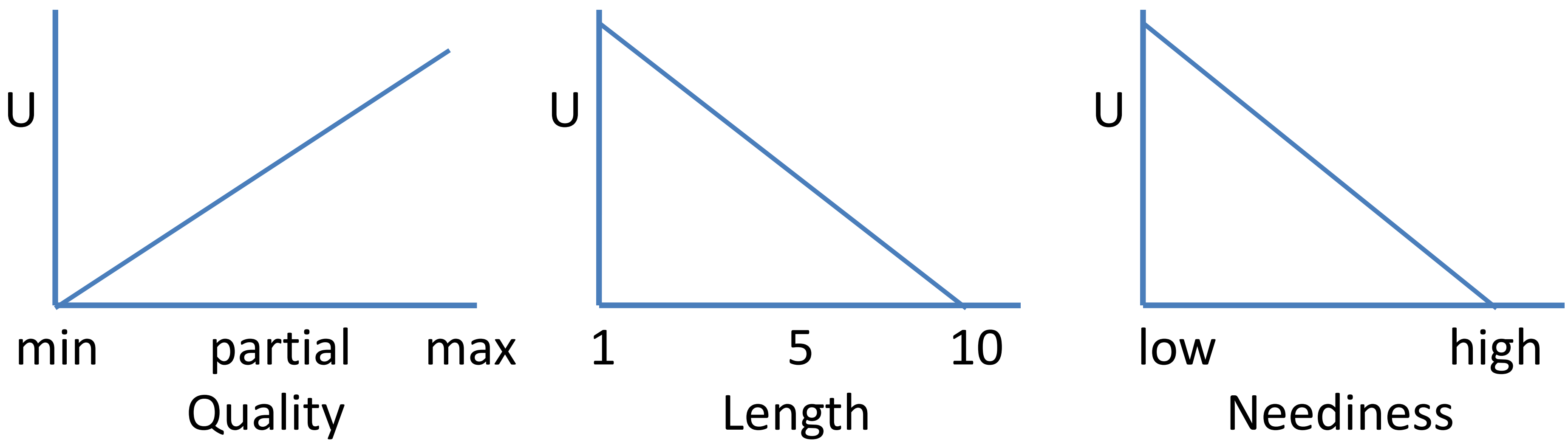
Intelligent Assistance as a Planning Problem



- Interaction preferences
- Reward function (help utility)
- Policy

Preferences as a Utility Function

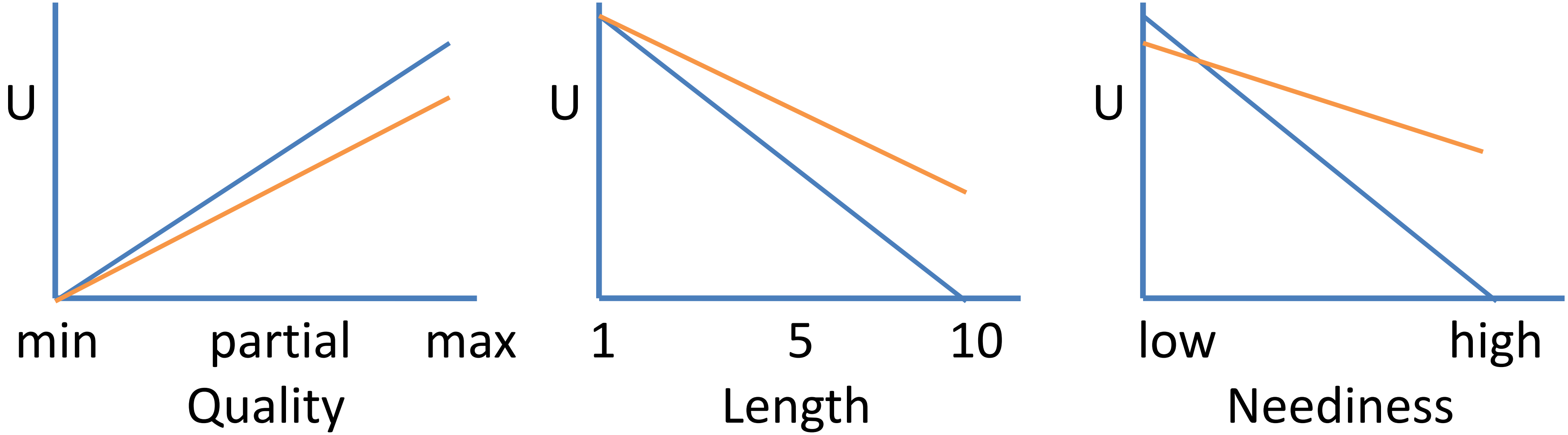
- Utility function for individual users
- Higher utility = "happier"



- Additive decomposition?

Preferences as a Utility Function

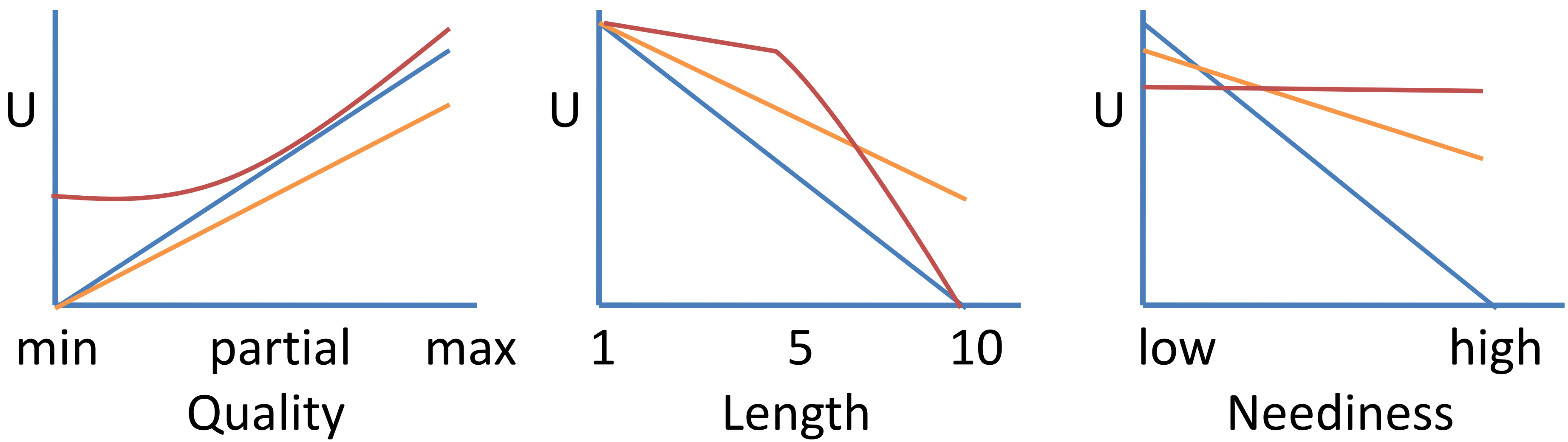
- Utility function for individual users
- Higher utility = “happier”



- Additive decomposition?

Preferences as a Utility Function

- Utility function for individual users
- Higher utility = "happier"



- Additive decomposition?

Background on Preferences

- Outcomes, O
- Utility function, $u: O \rightarrow Reals$
  - $u(o_i) > u(o_j)$ iff o_i is preferred to o_j
  - $u(o_i) = u(o_j)$ iff user is indifferent between o_i, o_j
  - o^- is best outcome s.t. $u(o^-) = 1$
  - o_- is worst outcome s.t. $u(o_-) = 0$
- **Strength** of preferences

Value and Costs of Suggestions

- Toolbar, *t*
- Highlighting goal, *g*
  - Complexity of goal
- Quality of toolbar, $Q(t|g) = \max_i Q(i|g)$
  - Quality of icon, $Q(i|g)$
- Neediness, $N(g)$
- Length, $L(t)$
- Suggestion utility, $\boldsymbol{U(N,L,Q)}$

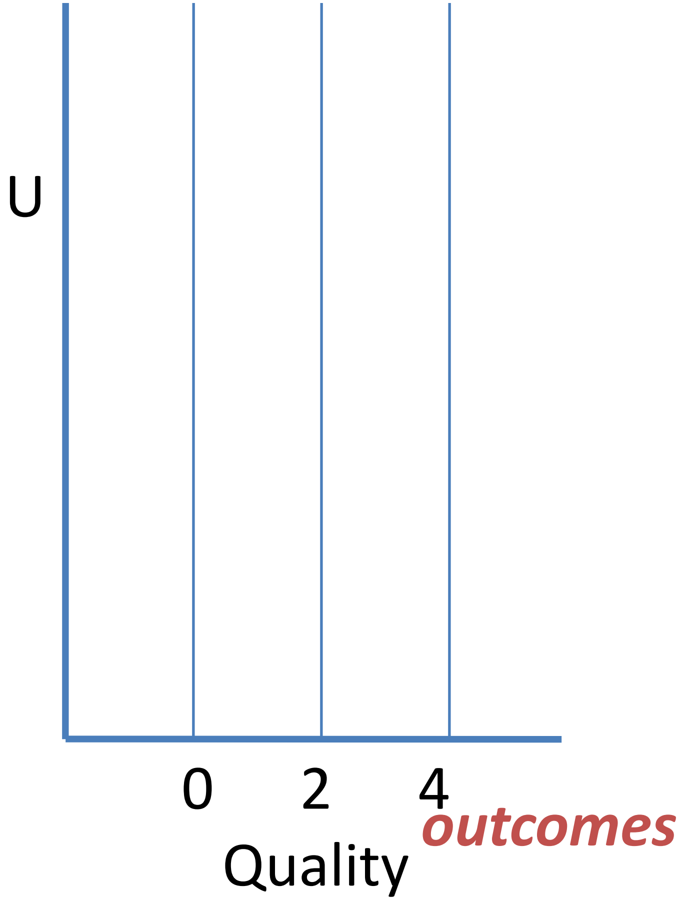
Preference Elicitation

- Standard gamble
  - $SG(pr) = [pr,o^{-}; 1\text{-}pr,o_{-}]$
  - Expected utility of SG = pr
- Standard gamble query
  - Alternative A: SG(pr)
  - Alternative B: o_i
  - Response: pr

Query Type	Question	Range of Responses
$SGQ(pr,o_i)$	What is *pr* s.t. $SG(pr) = o_i$?	$pr \in [0,1]$
$Bound(pr,o_i)$	Given *pr*, is $SG(pr) > o_i$?	Yes/No

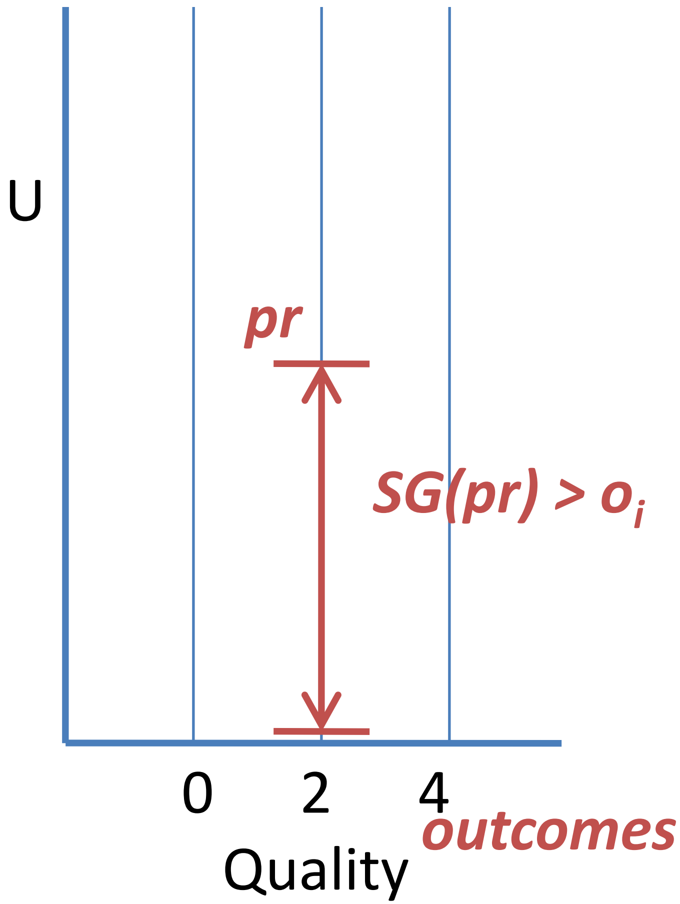
Bound(pr,o_i) in Theory

- Constraints allow incremental refinement



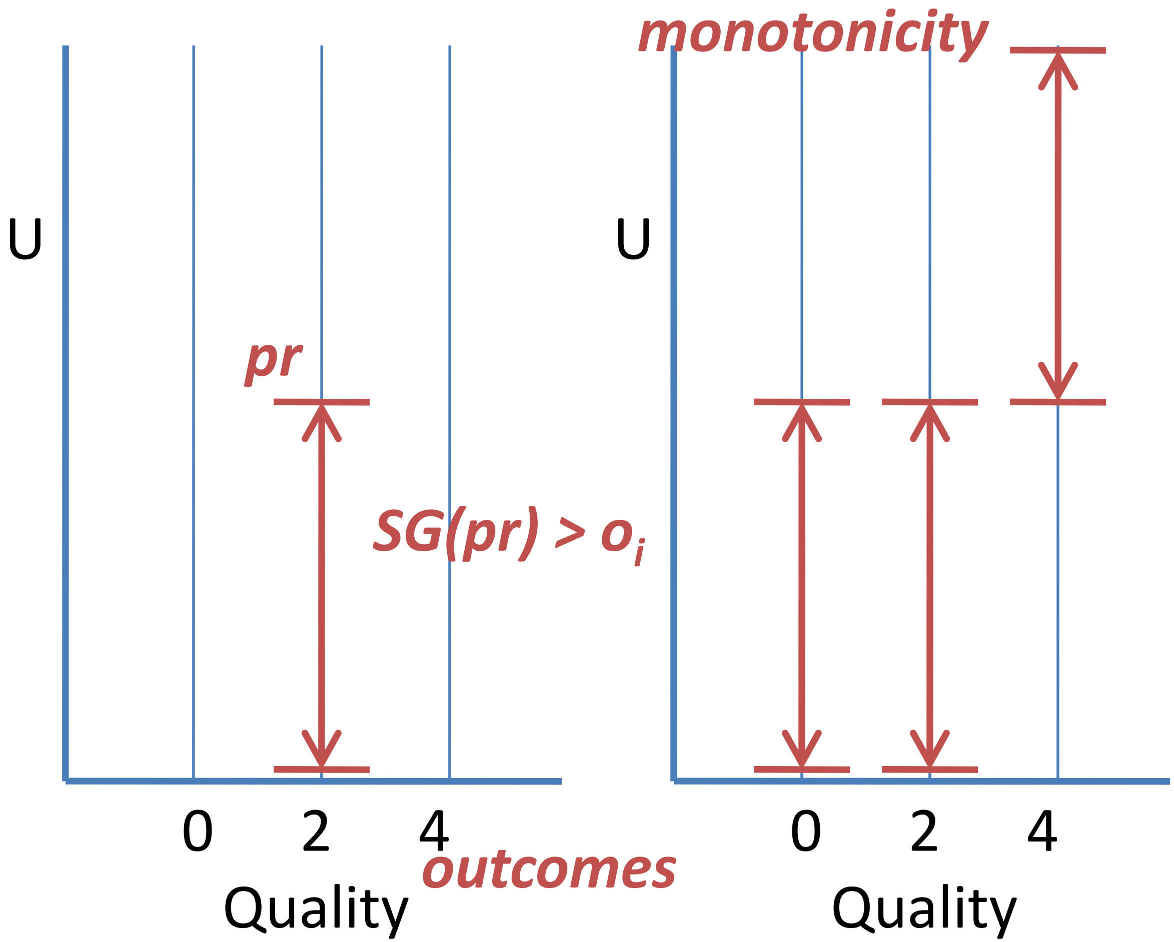
Bound(pr,o_i) in Theory

- Constraints allow incremental refinement



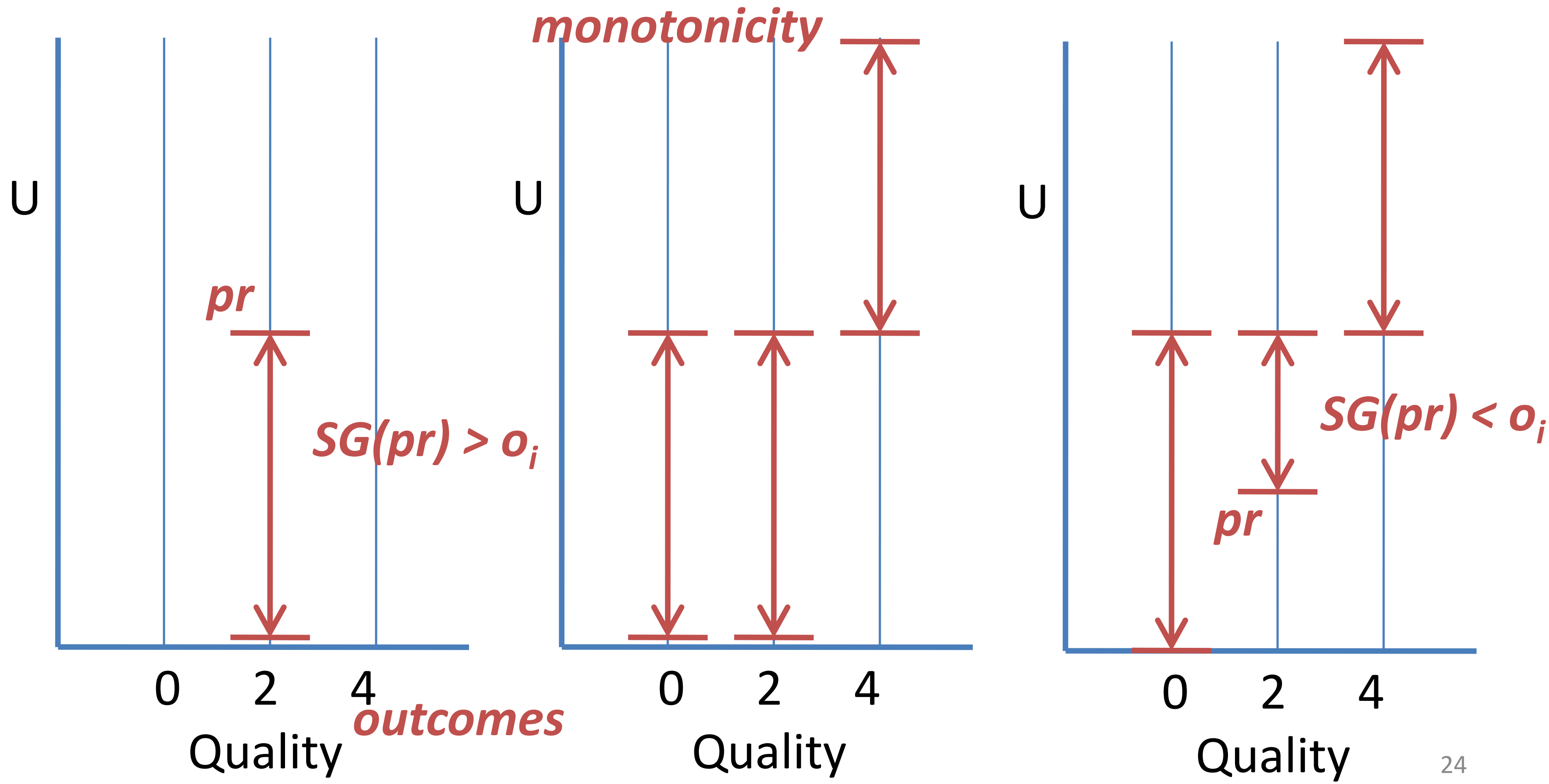
Bound(pr,o_i) in Theory

- Constraints allow incremental refinement

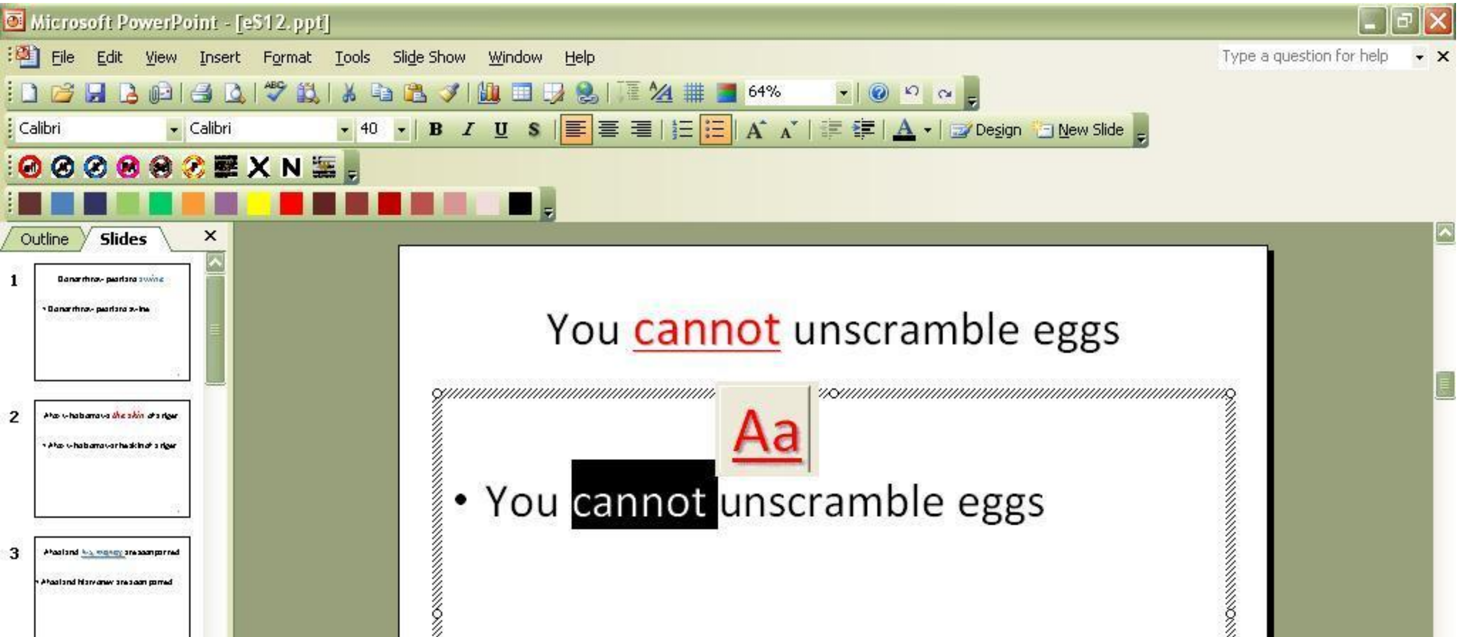


Bound(pr,o_i) in Theory

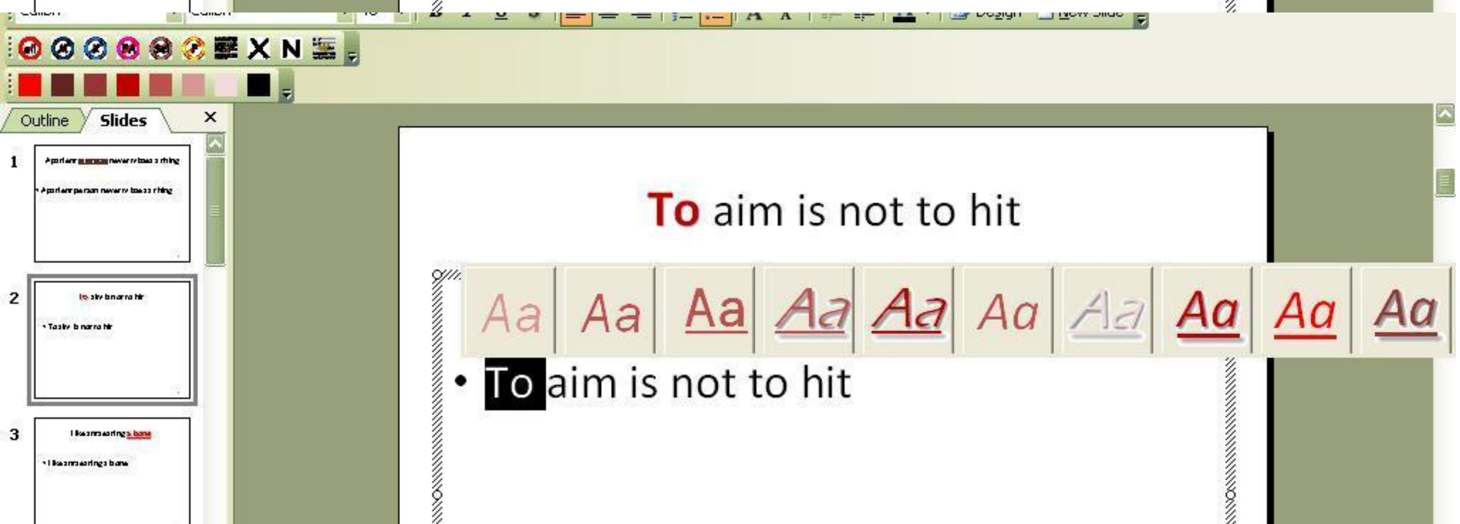
- Constraints allow incremental refinement



Bound(pr,o_i) in Practice



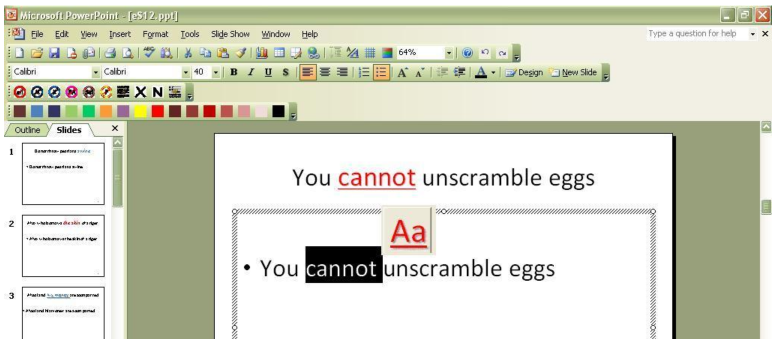
pr%

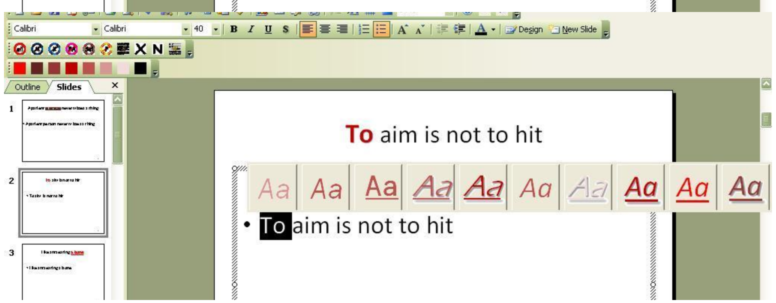


1-pr%

Bound(pr,o_i) in Practice

pr%

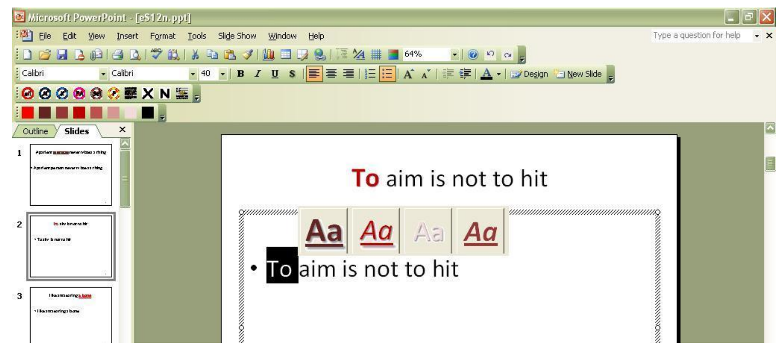




1-pr%

>

100%



?

Practical Difficulties

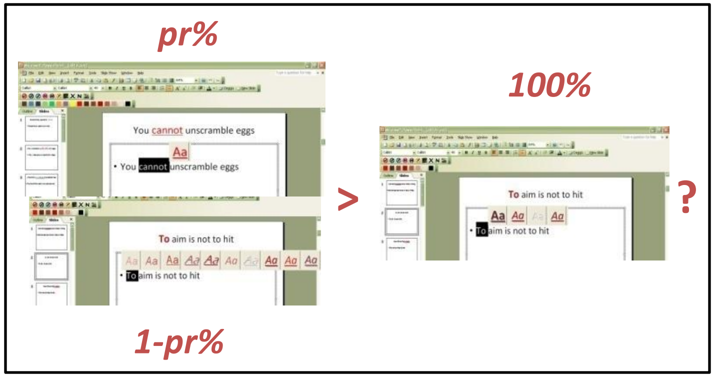
- Extremely informative, but…
  - Impossible to provide *pr* with confidence
  - SG (mixture of outcomes) difficult to interpret
  - Bound queries relatively easier
  - Difficult to distinguish as feasible regions get smaller
  - Sequential costs/benefits underestimated

Existing Approaches

- Qualitative approach of preferences rankings
  - Cannot quantify tradeoffs w.r.t. goal estimation
- Quantitative w. **conceptual** queries
  - Difficult to deploy in practice
    - AI: simulated users only
    - Psych: toy domains only
  - Assume people know what they like *a priori*
  - What people think they like ≠ what they actually like
- Ours: Quantitative w. ***Experiential*** queries

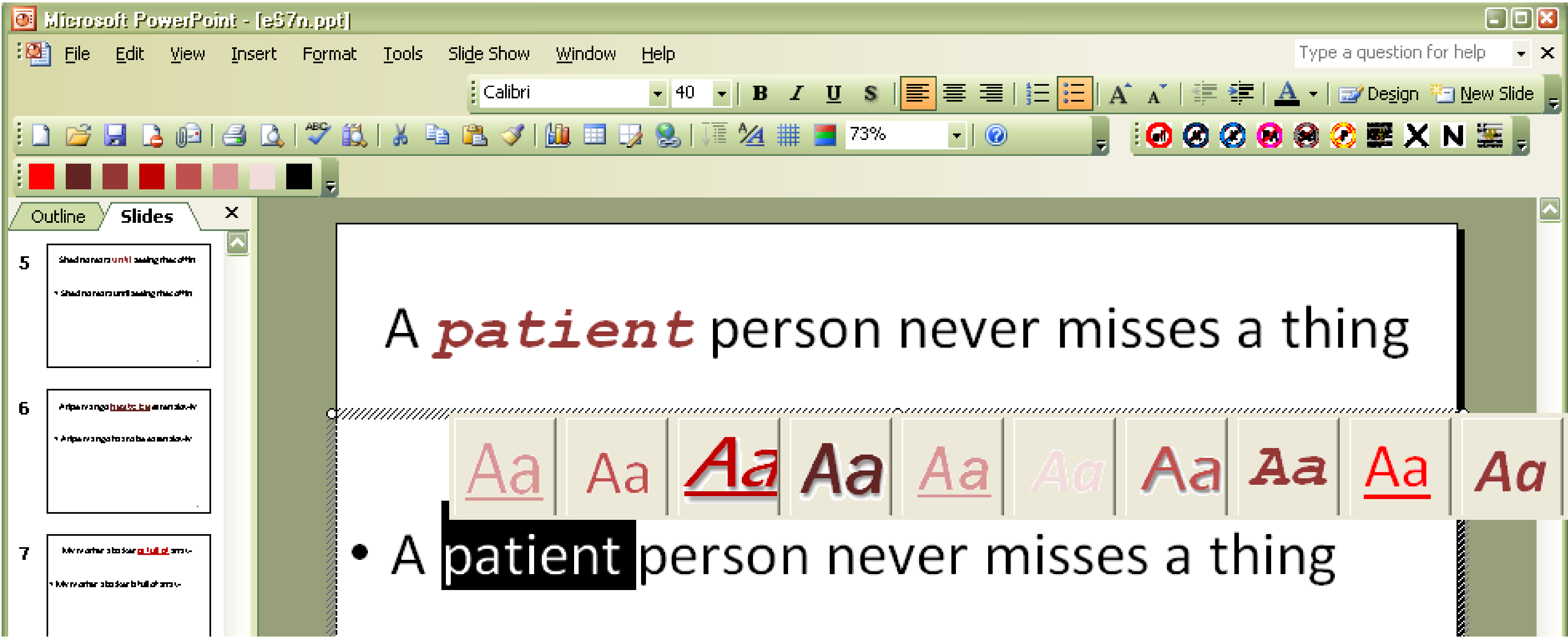
Experiential Queries

- Bound queries $B(pr,o_i)$: $SG(pr) > o_i$?
- U(N,L,Q) → U(outcomes)
  - O = interface configurations
- Experience via task completions
  - Simulate *pr* with *k* repeated tasks
  - Each query involves *2k* tasks
  - Discretize *pr* ∈ [0,.1,.2,…,1.0]



Experiment Set-up

- Controlled highlighting task in PowerPoint
- Simulated *N* into interface



Experiment Set-up

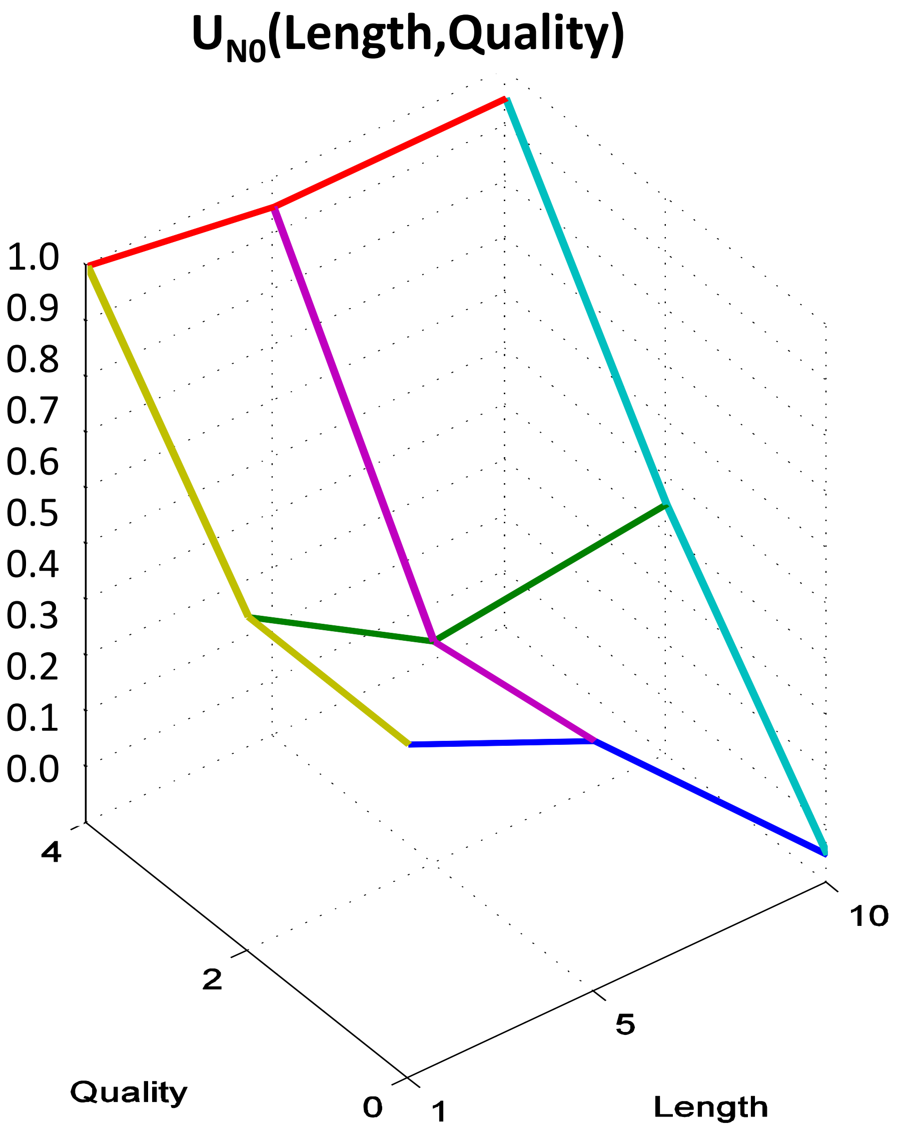
- Controlled highlighting task in PowerPoint
- Simulated *N* into interface
- Sampled from *U(N,L,Q)*
  - Neediness, *N = 0 (low), 1 (high)*
  - Length, *L = 1, 5, 10*
  - Quality, *Q = 0 (wrong), 2 (partial), 4 (perfect)*
  - o^{-} = N0,L1,Q4
  - o_{-} = N1,L10,Q0
- Treat options as adaptive vs. static system
- Elicited until small “feasible” regions (*pr ± 0.05)*

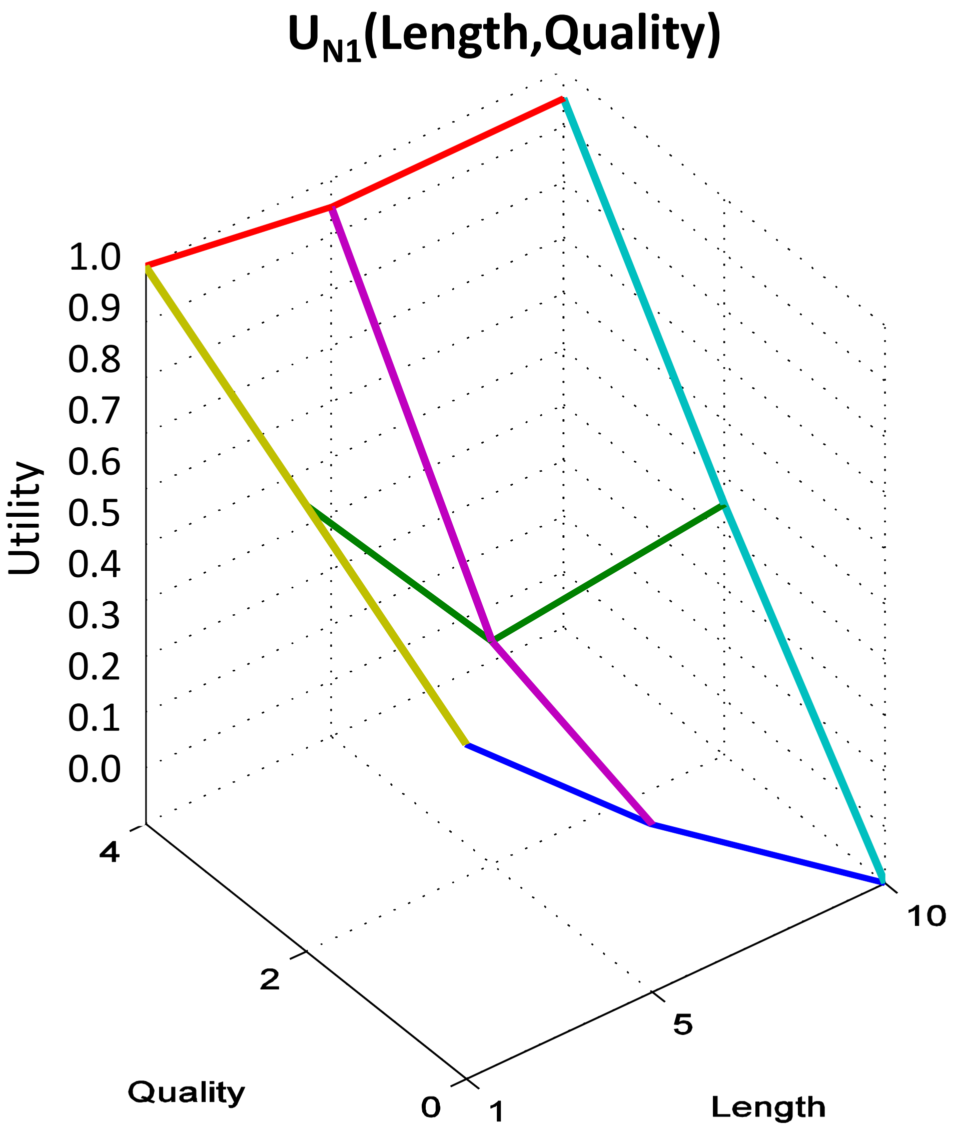
Experiential vs. Conceptual

- Conceptual
  - Imagine task completions
  - 13 participants
- Experiential
  - Carry out task completions
  - 8 participants

Structural Results

User 11 (midpoints)

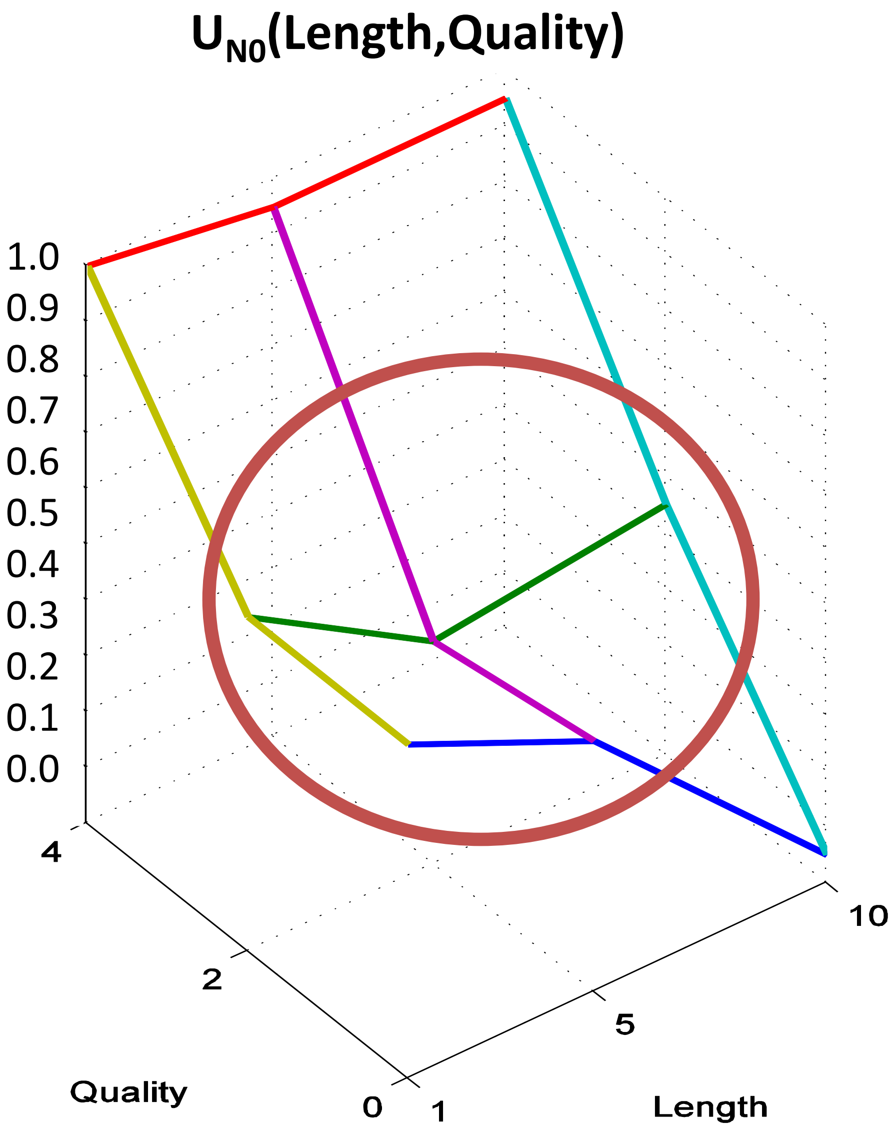


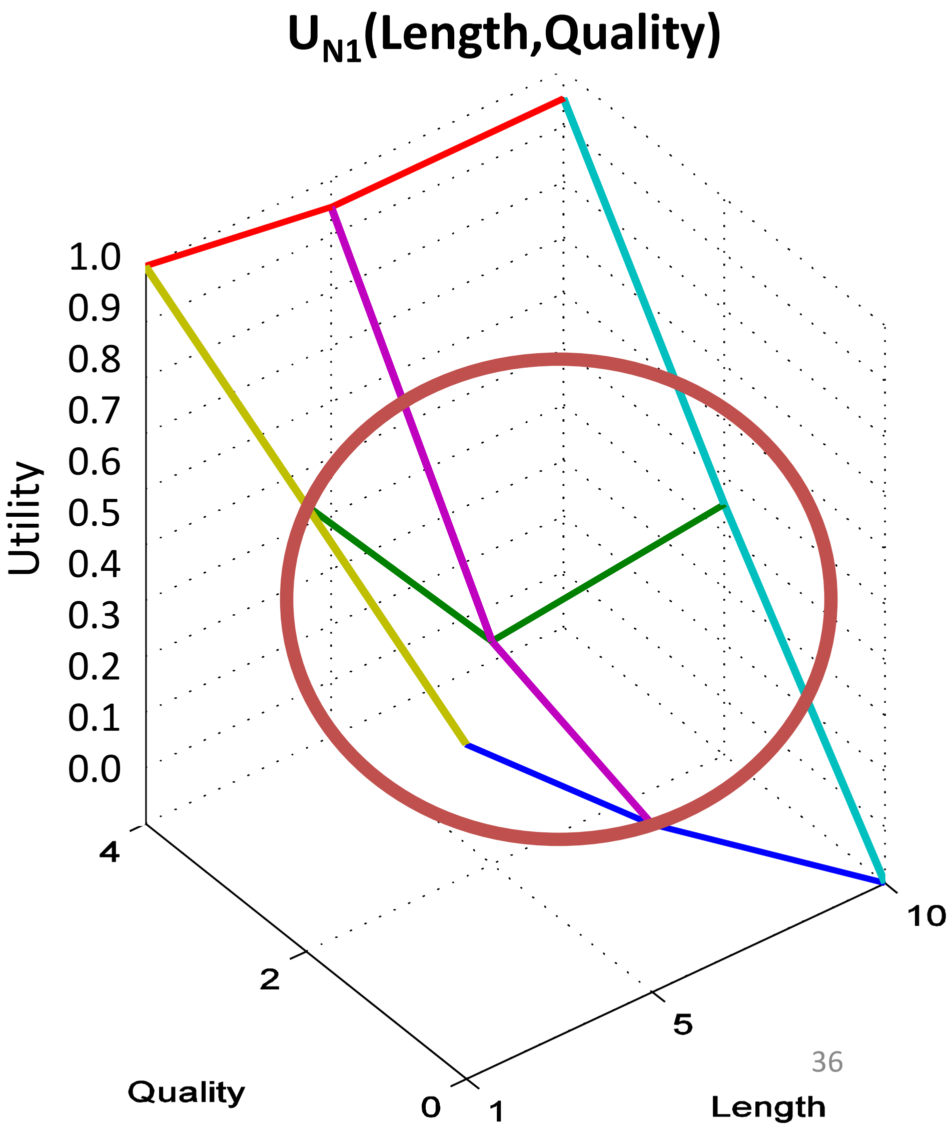


Structural Results

- Value of non-perfect help (Q2, even Q0)

User 11 – value in non-perfect help

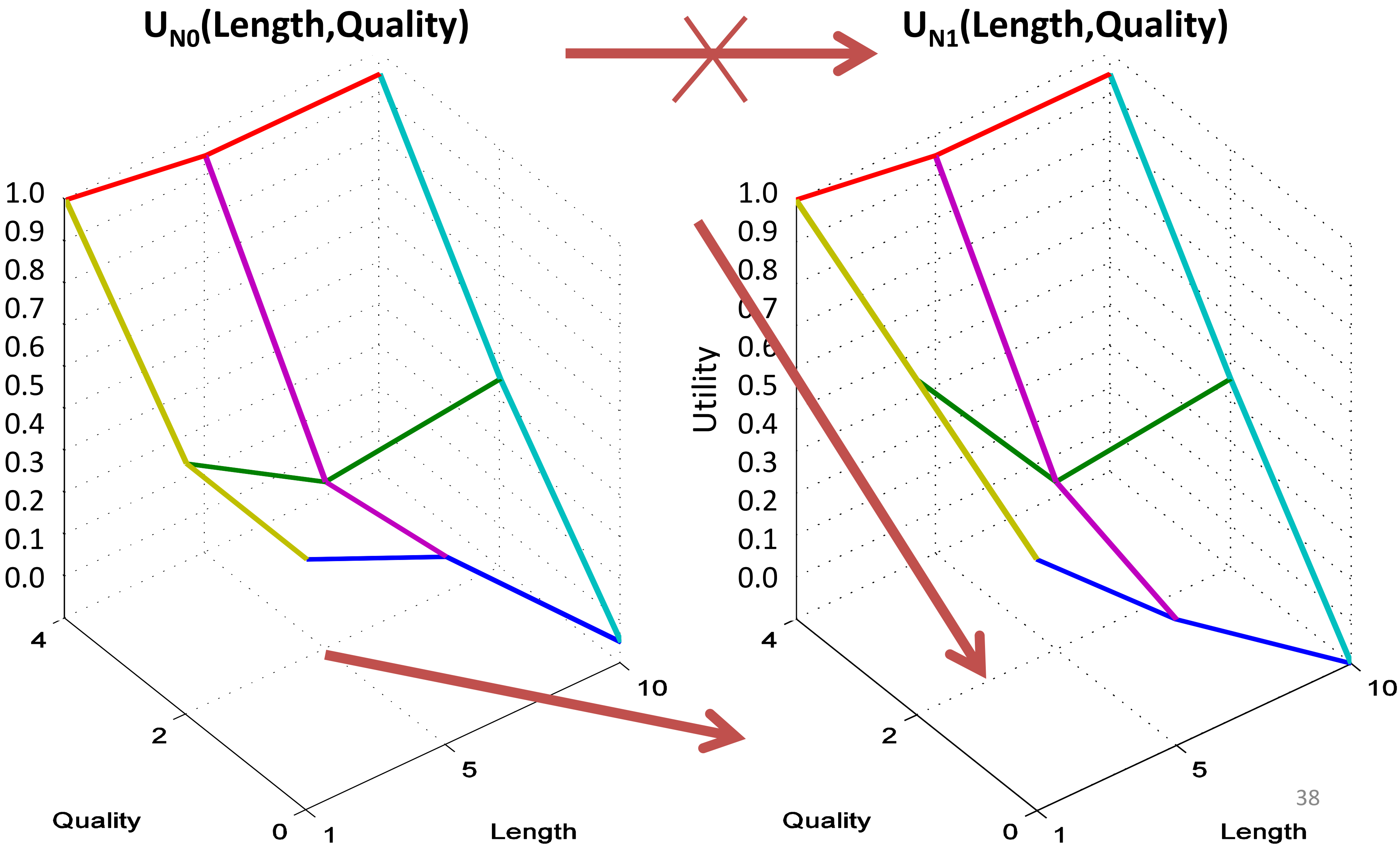




Structural Results

- Value of non-perfect help (Q2, even Q0)
- Monotonically non-increasing in *L*
- Monotonically non-decreasing in *Q*
- Variations in *N* (*user feature*)

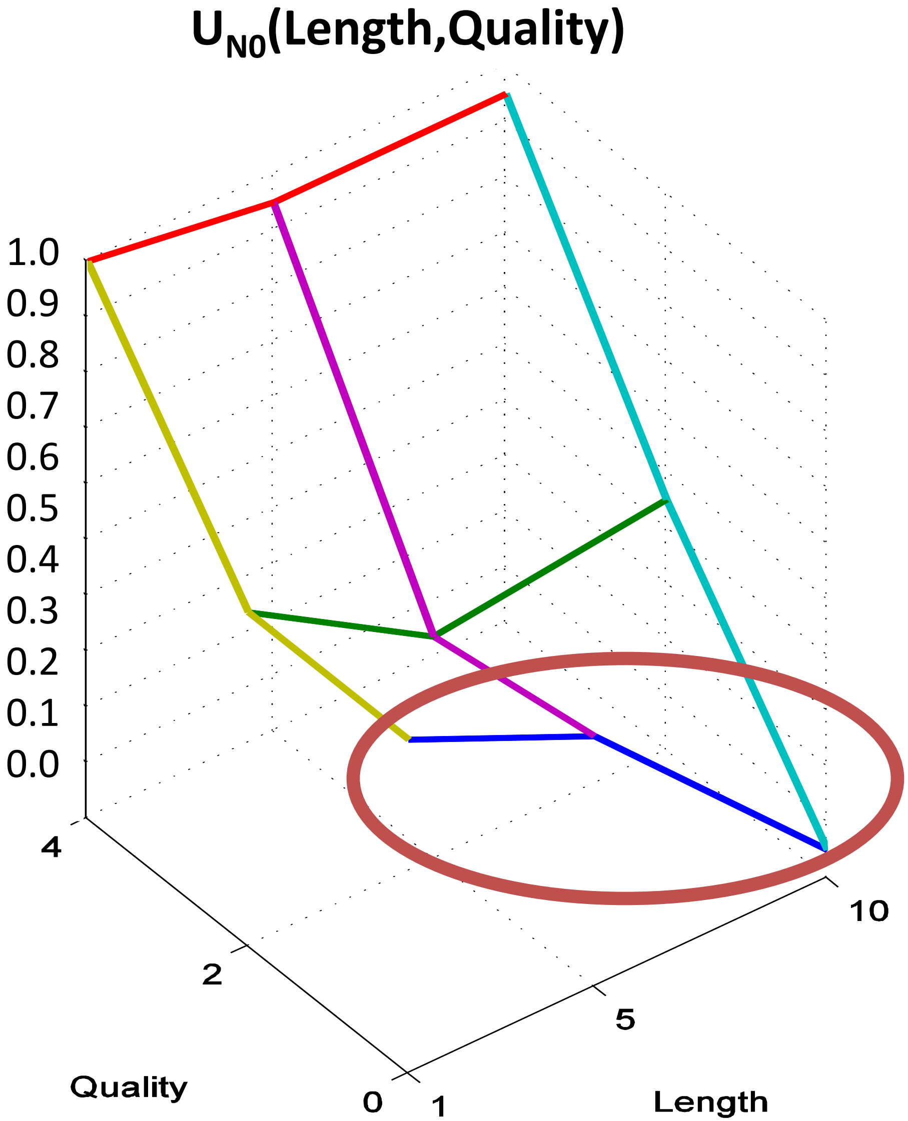
User 11 – monotonicity

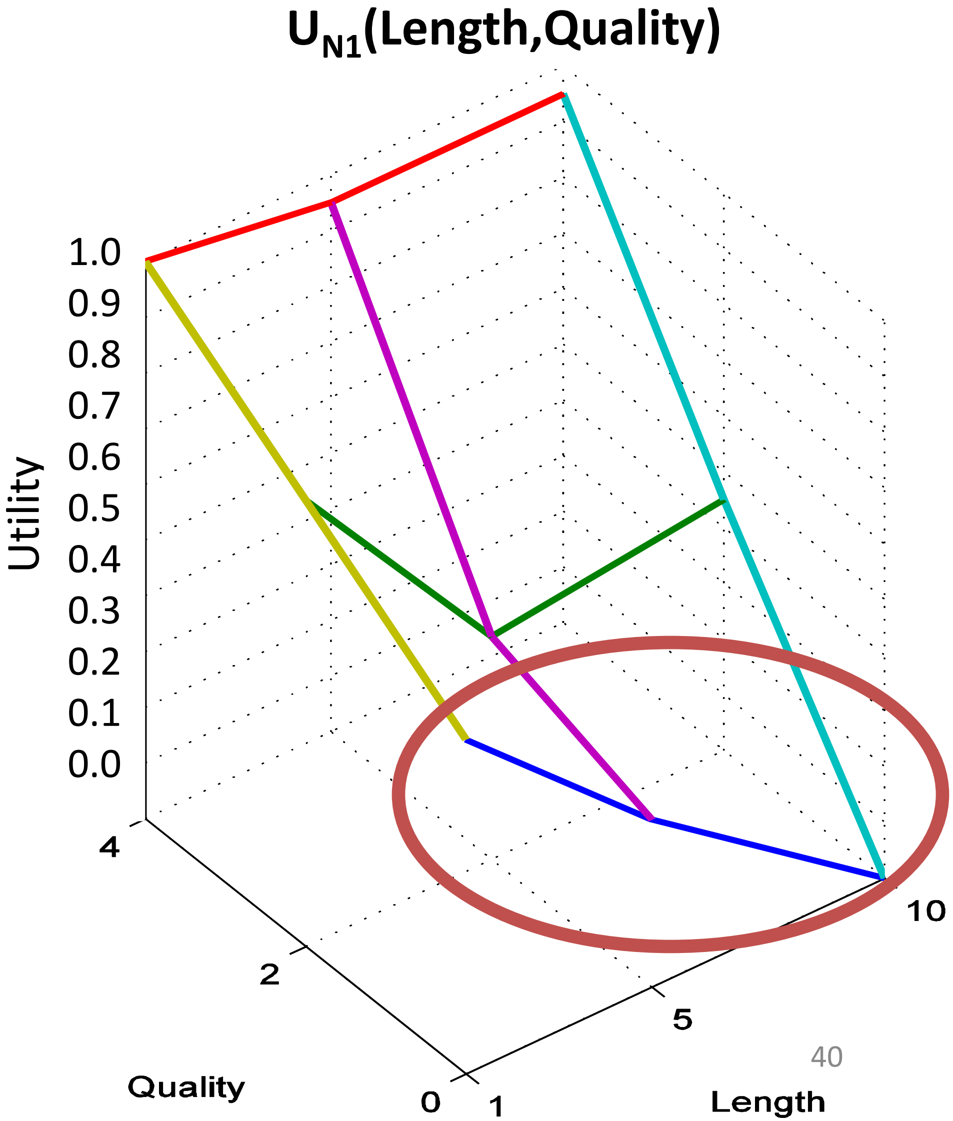


Structural Results

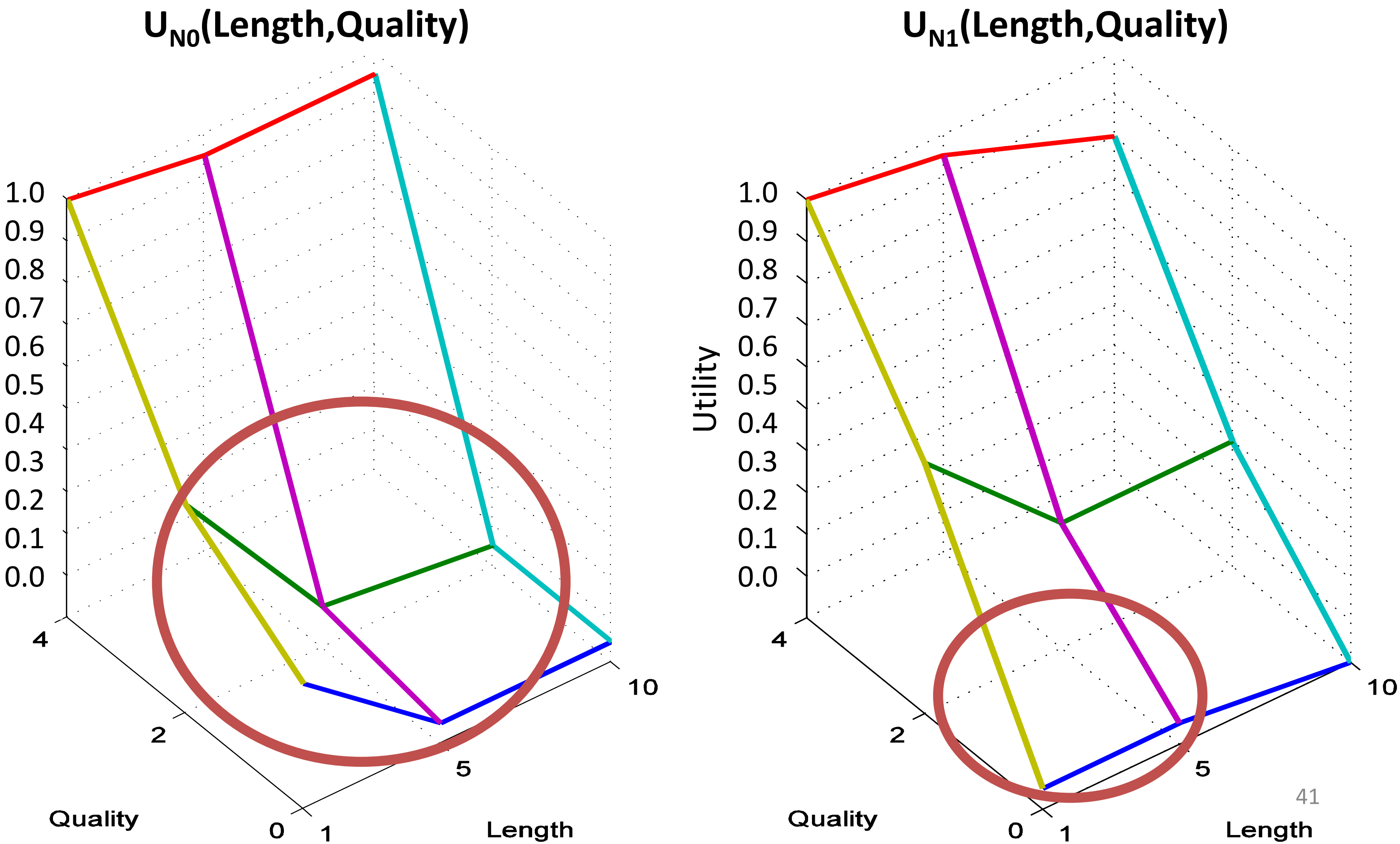
- Value of non-perfect help (Q2, even Q0)
- Monotonically non-increasing in *L*
- Monotonically non-decreasing in *Q*
- Variations in *N* (*user feature*)
- Curvature of partial functions

User 11 – curvature





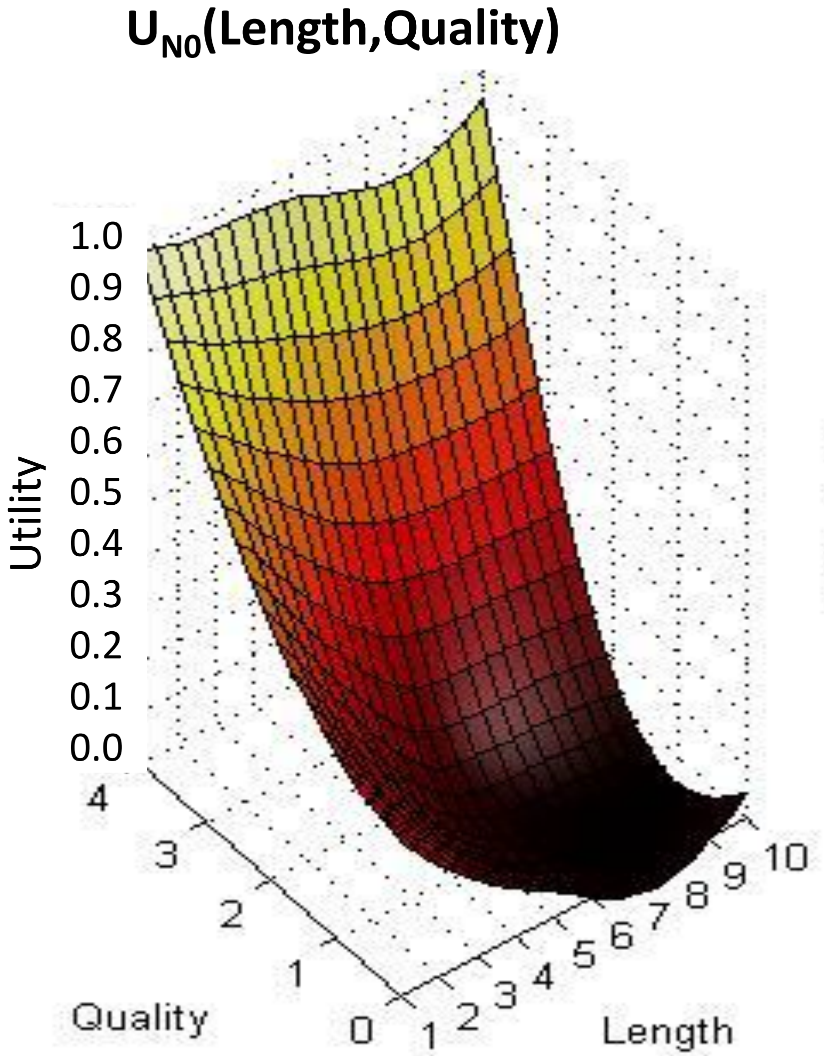
User 12 – differences in Q and L

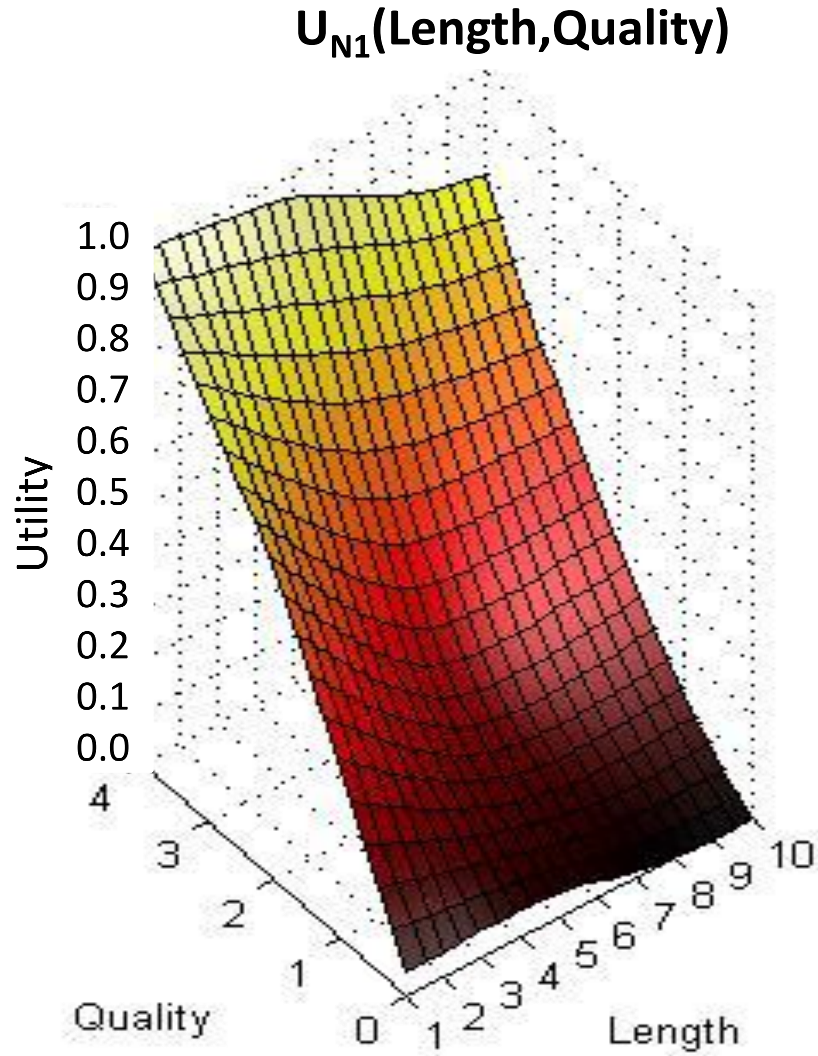


Structural Results

- Value of non-perfect help (Q2, even Q0)
- Monotonically non-increasing in *L*
- Monotonically non-decreasing in *Q*
- Variations in *N* (*user feature*)
- Curvature of partial functions
- **Non-additive decomposition**

Interpolated Utility Function





Experiential vs. Conceptual

- Methodological
- Quantitative

Experiential vs. Conceptual

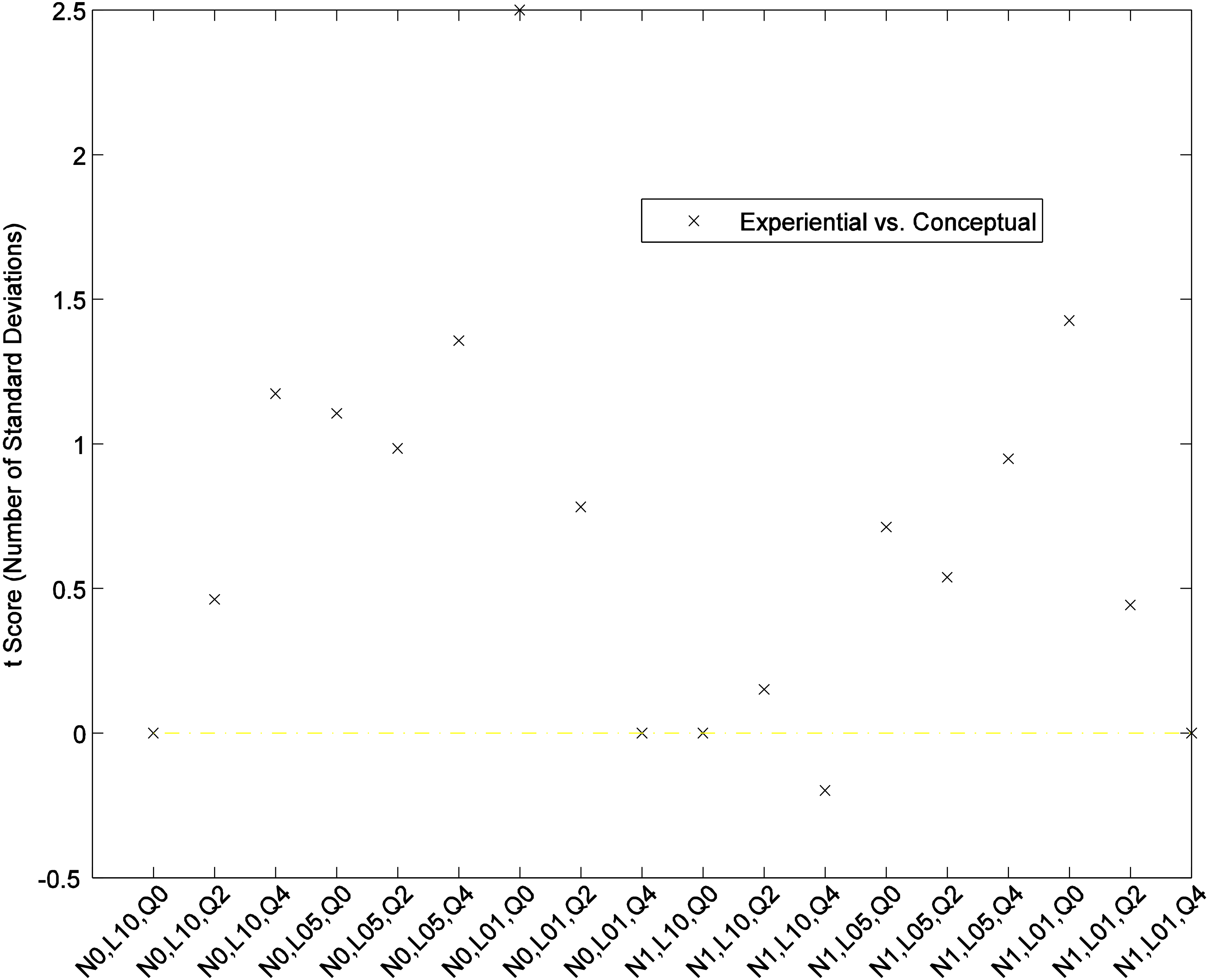
- Methodological

Experiential	Conceptual
2 hours	30 minutes
Easy to administer	Difficult to explain (mixture, repeated scenario)
Tired easily	Not easily tired
Generally consistent	Often inconsistent

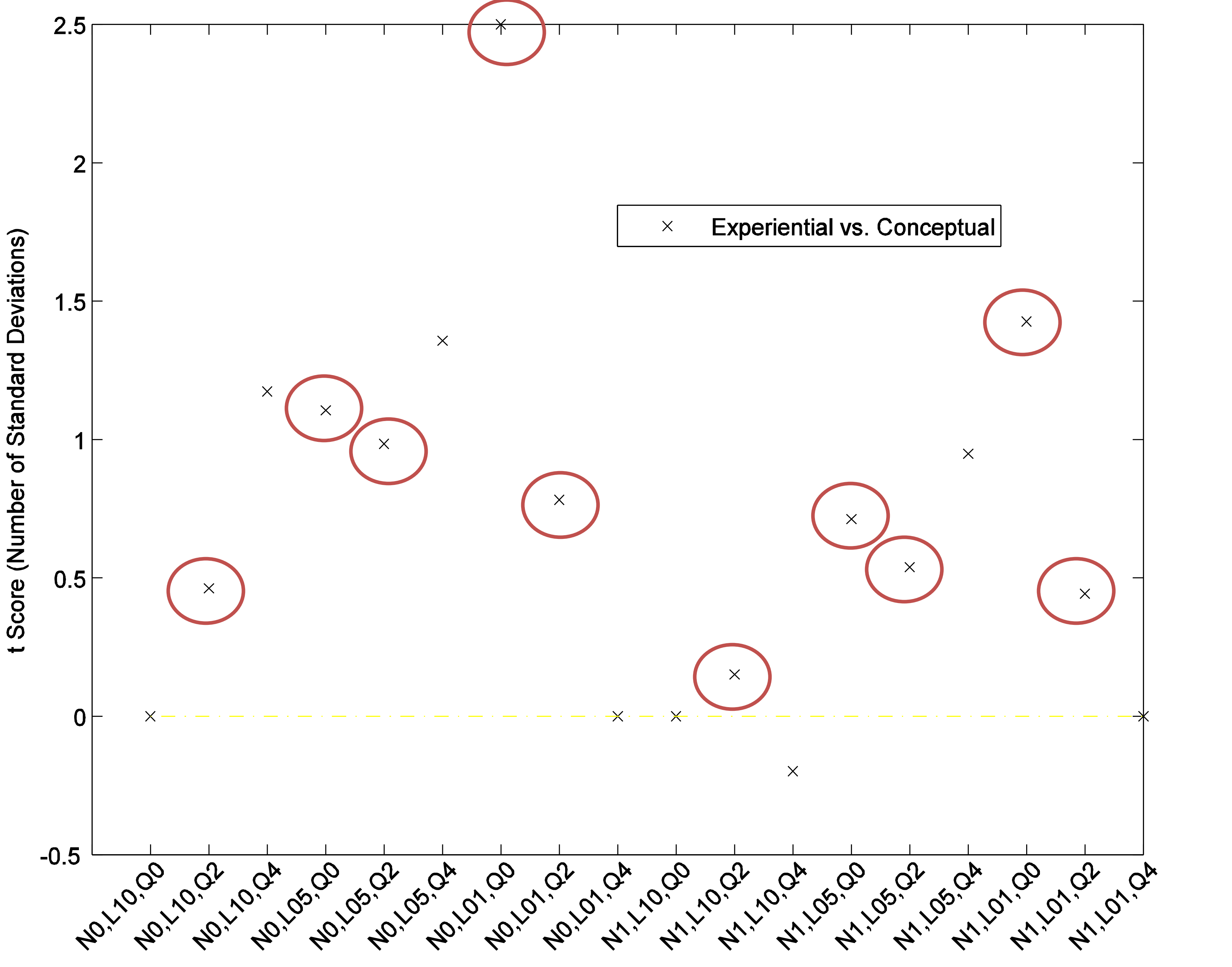
- Quantitative

Experiential vs. Conceptual

- Methodological
- Quantitative
  - H_0: experiential μ = conceptual μ
  - T^2 shows significance ($p < 0.01$)
    therefore, reject H_0
  - Component-wise *t* test with independent means



t Score (Number of Standard Deviations)
Experiential vs. Conceptual
2.5
2
1.5
1
0.5
0
-0.5
N0,L10,Q0
N0,L10,Q2
N0,L10,Q4
N0,L05,Q0
N0,L05,Q2
N0,L05,Q4
N0,L01,Q0
N0,L01,Q2
N0,L01,Q4
N1,L10,Q0
N1,L10,Q2
N1,L10,Q4
N1,L05,Q0
N1,L05,Q2
N1,L05,Q4
N1,L01,Q0
N1,L01,Q2
N1,L01,Q4



t Score (Number of Standard Deviations)
2.5
2
1.5
1
0.5
0
-0.5
Experiential vs. Conceptual
N0,L10,Q0
N0,L10,Q2
N0,L10,Q4
N0,L05,Q0
N0,L05,Q2
N0,L05,Q4
N0,L01,Q0
N0,L01,Q2
N0,L01,Q4
N1,L10,Q0
N1,L10,Q2
N1,L10,Q4
N1,L05,Q0
N1,L05,Q2
N1,L05,Q4
N1,L01,Q0
N1,L01,Q2
N1,L01,Q4

Experiential vs. Conceptual

- Methodological
- Quantitative
  - H_0: experiential μ = conceptual μ
  - T^2 shows significance ($p < 0.01$)

    therefore, reject H_0
  - Component-wise *t* test with independent means
    - Experience enables user to perceive value of automated help in repeated scenarios

Improving Experiential Elicitation

- Reduce time (thus, reduce effort)
- Training session
  - Familiarity with interface and help parameters
  - **Primed** condition
  - 9 participants
- Training + 5 experiential queries
  - **Primed+** condition
  - 8 participants

Primed/Primed+ vs. Conceptual

- Methodological
- Quantitative

Primed/Primed+ vs. Conceptual

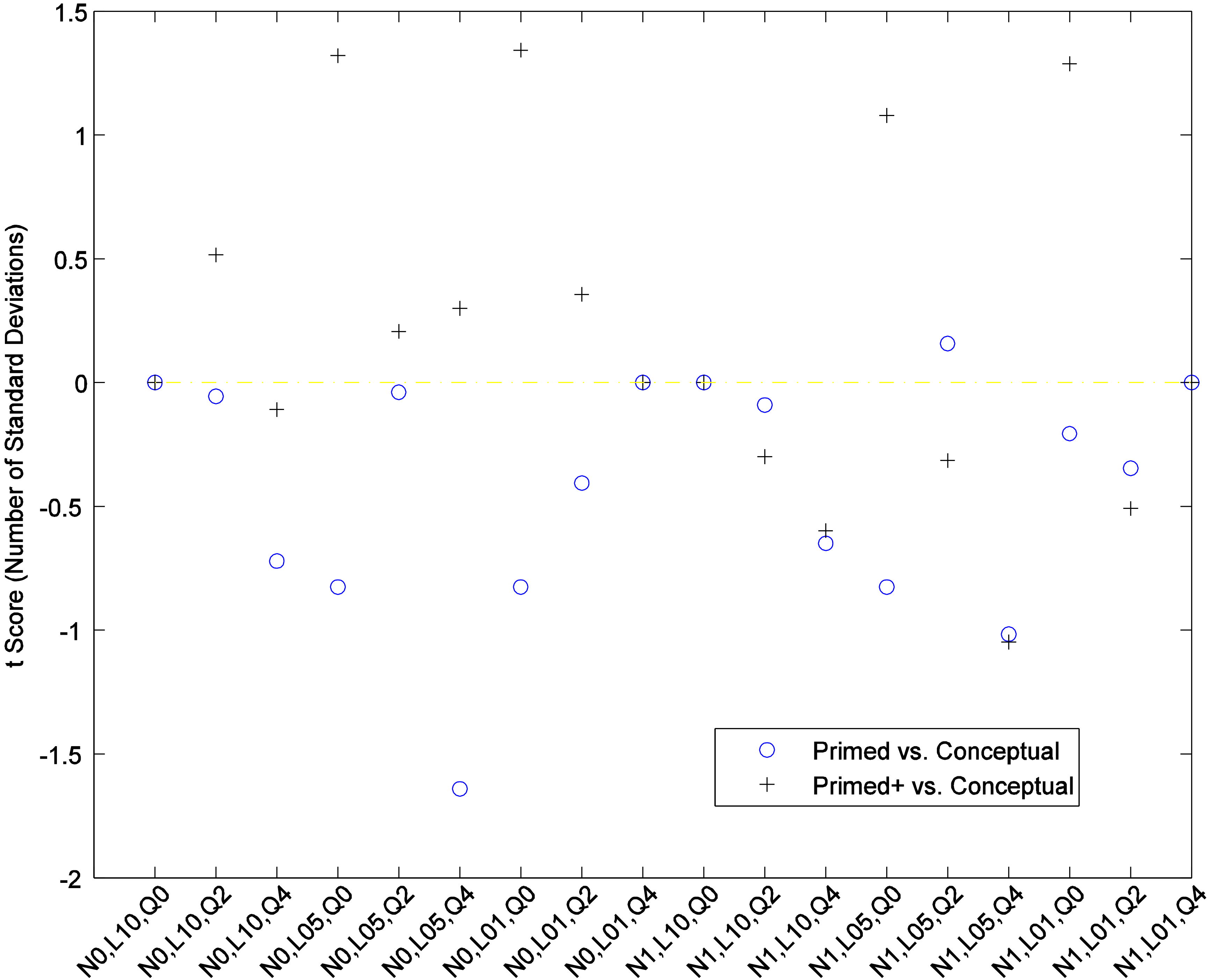
- Methodological

Primed	Primed+	Conceptual
30 minutes	60 minutes	30 minutes
Easy to administer	Easy to administer	Difficult to explain (mixture, repeated scenario)
Not easily tired	Not easily tired	Not easily tired
Often inconsistent	Experiential queries primed future responses	Often inconsistent

- Quantitative

Primed/Primed+ vs. Conceptual

- Methodological
- Quantitative
  - $H1_0$: primed μ = conceptual μ
  - $H2_0$: primed+ μ = conceptual μ
  - T^2 shows significance (H1:$p < 0.01$; H2:$p < 0.05$) therefore, reject $H1_0$ and $H2_0$
  - Component-wise *t* test with independent means



t Score (Number of Standard Deviations)
1.5
1
0.5
0
-0.5
-1
-1.5
-2
N0,L10,Q0
N0,L10,Q2
N0,L10,Q4
N0,L05,Q0
N0,L05,Q2
N0,L05,Q4
N0,L01,Q0
N0,L01,Q2
N0,L01,Q4
N1,L10,Q0
N1,L10,Q2
N1,L10,Q4
N1,L05,Q0
N1,L05,Q2
N1,L05,Q4
N1,L01,Q0
N1,L01,Q2
N1,L01,Q4
Primed vs. Conceptual
Primed+ vs. Conceptual

Primed/Primed+ vs. Conceptual

- Methodological
- Quantitative
  - $H1_0$: primed μ ≠ conceptual μ
  - $H2_0$: primed+ μ ≠ conceptual μ
  - T^2 shows significance (H1: $p < 0.01$; H2: $p < 0.05$) therefore, reject $H1_0$ and $H2_0$
  - Component-wise *t* test with independent means
    - Primed+ approaches Experiential

Contributions

- **Experiential elicitation** for interface customization
  - Uses bound queries with real users
  - Provides repeated experience
- Clear differences between experiential vs. conceptual elicitation
- **Primed+** as an efficient approximation to experiential elicitation

Future Work

- Learn parametric form for *U(N,L,Q)*
  - Quadratic in *L* and *Q*?
  - More data
- Model general utility function
  - Beyond savings/processing
  - Occlusion, bloat, disruption, interruption, etc.
- Understand experiential "affordance"
  - User expectations
  - Richer domain (more attributes and outcomes)