Automatic Service Selection in Dynamic Wireless Network Environments

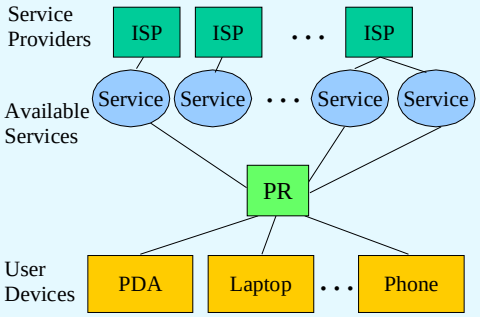
George Lee Peyman Faratin Steven Bauer John Wroclawski
{gjl,peyman,bauer,jtw}@lcs.mit.edu

Changing the Market for Wireless Network Access

Today:
- Few service providers
- Few service choices
- Users locked in to long−term contracts

Our Goal:
- Many local and regional providers
- Many diverse services
- Ability to switch between services easily, rapidly, and cheaply

The Personal Router (PR)



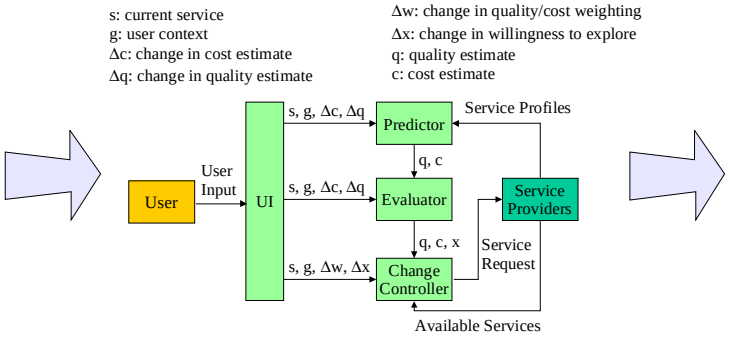
- Users need to discover and choose services
- Service providers need to advertise services
- We propose a Personal Router to automatically locate and select a service for the user

The Service Selection Problem

- The PR must intelligently and unobtrusively choose the right service for the user
- Many services available:
  - Advertised using a *service profile*
  - Services differ in quality and cost
  - Set of available services can change dynamically
- Correct service depends on user:
  - User context (e.g. running applications)
  - Quality vs. cost tradeoff
  - Willingness to explore new services

A Dynamic Learning Agent

selects network services based on learned user preferences



User Interface

- Intuitive and unobtrusive
- User inputs:
  - Four buttons: *better*, *cheaper*, *explore more*, *explore less*
- Translates from inputs to
  - Δq: change in quality estimate
  - Δc: change in cost estimate
  - Δw: change in quality/cost weighting
  - Δx: change in willingness to explore

Service Evaluator

- Learns user perceived quality and cost of services
- Inputs:
  - s, g, Δq, Δc from UI
  - q, c estimates from predictor for new services
- Outputs:
  - q, c estimates for change controller based on EWMA
  - Willingness to explore x for change controller

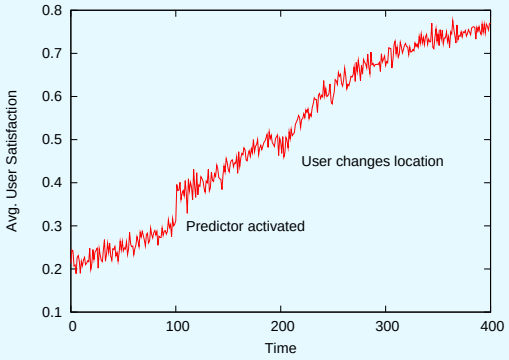
Service Change Controller

- Decides when to change and what service to select
- Inputs:
  - s, g, Δw, Δx from UI
  - q, c, x from evaluator
- Outputs:
  - Stochastically requests network services based on utility function of quality and cost estimates

Service Value Predictor

- Evaluates services not yet experienced by the PR
- Inputs:
  - s, g, Δq, Δc from UI
  - Service profiles from service providers
- Outputs:
  - q, c estimates learned using neural networks

Results and Evaluation



- Simulations show that the PR can learn user preferences even when services change and descriptions are incomplete
  - User satisfaction of a simulated user averaged over 1000 trials
  - User satisfaction depends on service profile and an unobservable hidden feature
  - 100 services simultaneously available
  - Available services change at time 200
- Presently evaluating the PR with experiments involving real users and networks
  - Comparing PR to optimal policy and manual selection
  - Testing ability of PR to make selections for web browsing when services change

Future Work

- Inferring user state from behavior
- Dealing with rapidly changing and complex user contexts
- Modeling complex and uncertain network environments

ANA
Advanced Network Architecture
Computer Science and Artificial Intelligence Lab
Massachusetts Institute of Technology