

# OneClick: A Framework for Capturing Users' Network Experiences

Cheng-Chun Tu<sup>1</sup>, Kuan-Ta Chen<sup>2</sup>, Yu-Chun Chang<sup>3</sup>, and Chin-Laung Lei<sup>3</sup>  
<sup>1</sup>Stony Brook University, <sup>2</sup>Academia Sinica, <sup>3</sup>National Taiwan University



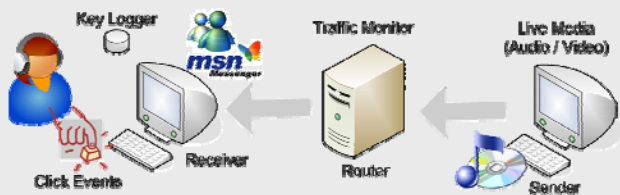
## Motivation

How to capture users' perceptions when they are using network applications?

Network Application	Delay Jitter	Loss Rate	User Satisfaction
1	Small	High	?
2	Large	Low	?

Which network configuration is better from users' perception?

## Network Topology



Router:

- Automatically change the network settings every 10 seconds
- For each setting, randomly set the network delay within (0, 2000) ms and loss rate within (0%, 20%).

Sender & Receiver:

- A song is played by the sender host, transmitted to the receiver via AIM or MSN Messenger.

## Contribution

- Present a lightweight, non-intrusive, and efficient framework called OneClick to learn about users' experiences in using network applications
- Demonstrate the effectiveness of assessing the users' perceptions under different network factors.

## Methodology

### User experiments

- Ask users to **click a button** whenever you feel dissatisfied with the quality
- Users do not need to be well-trained to participate as only an intuitive click action is required.

### Data analysis

- Apply **Poisson regression** to model the relationship between network factors and the click rate (average number of times the subject clicks the button in one second)
- Predictor: **click rate**
- Dependent variable: **network factors**

Assume the click rate is  $C(t)$  and the network factors are  $N_1(t), N_2(t), \dots, N_k(t)$  at time  $t$ . Then, the Poisson regression equation is:

$$\log(C(t)) = \alpha_0 + \alpha_1 N_1(t) + \dots + \alpha_k N_k(t), \quad (1)$$

where  $\alpha_i$  denotes the Poisson regression coefficients, which are estimated by the maximum likelihood method

### Users' delay response

- Problem: users may unintentionally delay the click actions after they become aware of the degraded application quality
- Solution: shift the click event process and search for the average delay time  $d_{avg}$  by **fitting the regression model** for network factors and click event processes with different time lags
- $d_{avg}$  is computed as:

$$\text{argmin}_d \{ \text{deviance of (1) by replacing } C(t) \text{ with } C(t+d) \}$$

After obtaining the best model fit, we compute the **expected click rate** as:

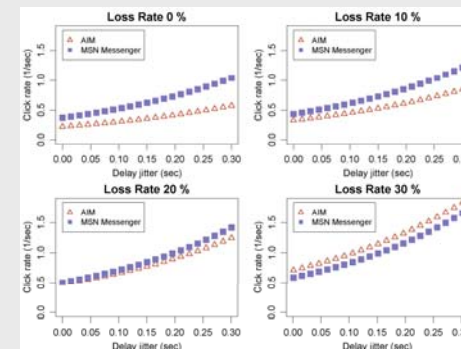
$$\exp(\beta_0 + \beta_1 N_1 + \dots + \beta_k N_k),$$

where  $\beta_i$  is the Poisson regression coefficients by fitting  $C(t + d_{avg})$  with  $N_i, 1 \leq i \leq k$

## Experimental Setup

- A song is played by the sender host, transmitted to the receiver via AIM or MSN Messenger, and played on the subject's earphone attached to the receiver host
- Subjects: three computer science students to perform the experiments, where each experiment lasts for 6 minutes

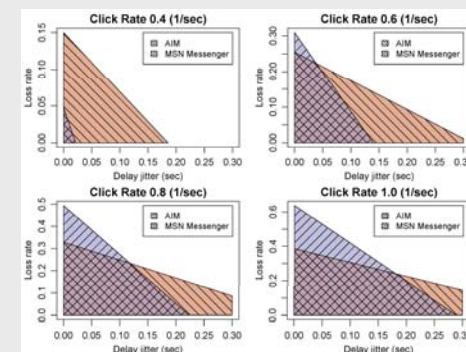
## Result: Expected Click Rate



- Loss rate < 20%: MSN leads to a higher click rate
- Loss rate  $\geq$  30%: AIM leads to a higher click rate
- AIM is less tolerant of an extremely high packet loss rate than MSN Messenger, even though it is more tolerant at low packet loss rate

## Result: Comfort Regions

"comfort region": the set of network scenarios that leads to an expected click rate lower than a certain **threshold**



- Threshold = 0.4 or 0.6: AIM is better (larger comfort regions)
- Threshold = 0.8 or higher: Comparable
  - MSN Messenger is more tolerant of packet loss and AIM is more tolerant of network delay jitters



