A Contextual Framework for Adaptive User Interfaces

Modelling the Interaction Environment

Mateusz Dubiel*
Bereket Abera Yilma*
Kayhan Latifzadeh
Luis A. Leiva
name.surname@uni.lu
University of Luxembourg
Luxembourg

ABSTRACT

The interaction context (or *environment*) is key to any HCI task and especially to adaptive user interfaces (AUIs), since it represents the conditions under which users interact with computers. Unfortunately, there are currently no formal representations to model said interaction context. In order to address this gap, we propose a contextual framework for AUIs and illustrate a practical application using learning management systems as a case study. We also discuss limitations of our framework and offer discussion points about the realisation of truly context-aware AUIs.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; *HCI theory, concepts and models*; *Ubiquitous and mobile computing theory, concepts and paradigms.*

KEYWORDS

Adaptation; Interface Personalisation; Design; Interaction Context; Environment

1 INTRODUCTION

Research in Adaptive User Interfaces (AUIs) has provided empirical evidence about the importance of contextual information from the user's environment to aid personalisation of UIs [Shankar et al. 2007]. Although there are several works on context-aware AUIs that explore the role of said environment, this research area remains open. In practice, users evolve in a physical space together with other people and devices that often tend to influence the behaviour of the users. Additionally, a physical space by itself has certain rules that govern user behaviour and impose additional constraints on

---

*Both authors contributed equally to this research.

how users interact with computers, which ultimately determines how interfaces should be adapted. Nevertheless, classic approaches to AUIs focus on cues and signals solely collected from users to personalise interfaces, largely ignoring contextual information. In this work we propose a general framework which allows to incorporate the user's interaction context (or *environment*) in AUIs by design.

In order to provide contextual grounding for our framework we first briefly overview related work on AUIs (section 2). Then, in section 3, we formalise the problem by taking a systems theory perspective [Maier 1996], and present our proposed framework inspired by a classic reinforcement learning architecture [Sutton and Barto 1998; Watkins and Dayan 1992]. In section 4 we present a case study that illustrates a potential application of our framework. We finish with implications of our proposal and provide several points for discussion and future work (section 5).

2 RELATED WORK

Many studies in AUIs explore the role of user's feedback on interface personalisation. Some types of feedback collected include: using natural language programming to optimise behaviour of artificial agents and task disambiguation [Jiang et al. 2019]; introducing head-gesture analysis to facilitate hands-free interaction with Head Mounted displays [Yan et al. 2018]; enabling silent-speech interaction mode with a mobile device by analysing user's mouth opening degree [Sun et al. 2018]; utilising users' mouse behaviour to automatically modify layout of a website [Leiva 2012]; or using user body postures and physical activity to adjust the features of the surrounding environment such as ambient light, temperature or music [Wang et al. 2019].

Another line of research focused on system features that can be used for interface adaptation. For instance, Jiang et al. [Jiang et al. 2019] proposed a novel layout method that adds OR-constraints (ORC) to standard constraint-based layout specifications. The proposed method unifies grid layout and flow layout, offering new possibilities for flexible UIs that were not supported by any other layout method. In another study, Swearngin et al. [Swearngin et al. 2020] proposed the Scout system to help designers rapidly explore alternative interface layouts through mixed-initiative interaction with high-level constraints (e.g. semantic structure, emphasis, order) and design feedback.

2.1 AUI Frameworks

Previous formalizations of AUIs have approached the adaptation problem from different angles. One notable example in this area is the SUPPLE system [Gajos and Weld 2004] which framed UI adaptation as an optimisation problem. SUPPLE minimises user's effort by providing interface adaptations that meet device and user capabilities. However, SUPPLE did not account for the modeling of environment features.

Similarly, Bouzit et al. [Bouzit et al. 2017] formalised a design space for user interface adaptation, alas without modelling the environment. The proposed framework is based on Perception-Decision-Action cycle that is augmented by Learning-Prediction-Action, allowing for UI designs that are descriptive, comparative, and generative.

On the other hand, Abrahão et al. [Abrahão et al. 2021] proposed a conceptual reference framework for intelligent user interface adaptation with a set of conceptual adaptation properties that are crucial for model-based AUIs. While the authors mention an "Environmental Model" as a part of their "Context Model" that affects system adaptation, there is no formulation provided regarding the elements that define the environment and the role of their interactions.

Overall, while previous work on proposed conceptual modelling of the system, to the best of our knowledge, conceptual modelling of the environment has not been proposed yet. Moreover, while taxonomies of interaction environment exist there are currently no formalisation of this concept. Bearing in mind the importance of contextual information that can be obtained from interaction environment and benefits that it can be for development of AUIs, in the current paper we propose a conceptual framework for modelling the environment.

3 A CONTEXTUAL FRAMEWORK FOR AUIS

We propose that the environment of the user can be understood as a System-of-Systems (SoS) which contains sub-systems such as the user, the device which implements the adaptation or personalisation, other users and entities in the same physical space with the user that may have a direct or indirect influence on their behaviour or on the personalisation. Each of these sub-systems posses their own systemic properties; i.e. *components, objectives, relations, behaviour, structure, interface, environment, and functions* [Maier 1996]. Taking such a systemic view allows to better understand the coexisting entities, their inter-dependencies, as well as their influence on the environment and vice versa. Thus, we can formalise the problem of UI adaptation p as a function of the composing systems, given by

$$p = f(u, d, e) \tag{1}$$

where u is the user, d is the device to be adapted through its interface, and e is the environment potentially influencing the user as an independent system as well as containing subsystems itself, such as other people and devices that may have an impact on the user.

Adapting UIs often entails changing some functionalities such as information content, presentation layout, or distinctiveness of an interface to increase its personal relevance to the user. Doing so requires taking into account not only the user but also environmental constraints and potential influences imposed on the user.

This essentially means devising an efficient strategy to understand and reason about dynamic interaction responses of the user and, consequently, adapting the UI to the environmental changes. Implementing this, however, is not a trivial task as it requires to derive efficient representations of the user, the coexisting entities, and the environment's state from high-dimensional sensory inputs, and use these information sources to generalise past experiences to new situations in order to better adapt UIs.

Such types of challenging tasks are remarkably solved by humans and other animals through a harmonious combination of reinforcement learning (RL) and hierarchical sensory processing systems [Fukushima and Miyake 1982; Serre et al. 2005]. This in particular has inspired the development of several RL algorithms over the years, cf. Nguyen et al. [Nguyen et al. 2020]. Early RL algorithms were limited to domains in which useful features could be handcrafted, or to domains with fully-observed low-dimensional state spaces.

Recently, Deep Q-networks (DQNs) can learn successful policies directly from high-dimensional sensory inputs using end-to-end reinforcement learning [Mnih and Kavukcuoglu 2017]. DQNs have been tested in various complicated tasks and were able to outperform all previous RL algorithms [Silver et al. 2016, 2017]. DQNs have also enabled breakthroughs such as "AlphaGO" [Chen 2016] and"AlphaStar" [Arulkumaran et al. 2019], which have inspired recent work on AUIs in the context of linear menus [Todi et al. 2021]. These advancements demonstrate the potential of RL to build intelligent agents by giving them the freedom to learn by exploring their environment and make decisions to take actions that maximise a long term reward.

We believe that RL can be highly beneficial to AUIs, as it allows learning through exploration, unlike classic approaches and supervised methods, that often require large amounts of labelled data and are harder to train with continuous action spaces. Taking this inspiration, in the following we reformulate the goal of AUIs as an RL task by extending the high-level formalisation in Equation 1.

In classic RL, agents interact with their environment through a sequence of observations, actions, and rewards [Watkins and Dayan 1992]. At a given time, an agent takes an observation (i.e., information about the state of the environment) and takes an action that will maximise a long-term reward. The agent then observes the consequence of the action on the state of the environment and the associated reward. It then continues to make decisions about which actions to take in a way that maximises the cumulative future reward. This is done by learning action value function:

$$Q^*(s, a) = \max_\pi \mathbb{E} \left[\sum_{t \geq 0} \gamma^t r_t | s_0 = s, a_0 = a, \pi \right] \tag{2}$$

which is the maximum sum of rewards r_t discounted by γ at each time step t, achievable by a policy $\pi = p(a|s)$, after making an observation of s and taking an action a. This means that RL agents operate based on a policy π to approximate Q-values (state-action pairs) that maximise a future reward. Figure 1 illustrates the schematics of the different components in classic RL. We refer the reader to the work of Watkins and Dayan [Sutton and Barto 1998; Watkins and Dayan 1992] for the details on Q-learning and RL.
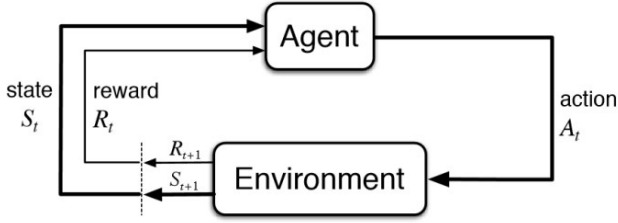
Figure 1: Classic Reinforcement Learning framework [Sutton and Barto 1998].

Adopting this to the context of AUIs, the agent corresponds to the device d which operates based on some policy π. The action taken by the device a_t at any time step t corresponds to selecting an optimal UI configuration layout denoted by $\mathbf{x} = (x_1, \ldots, x_N)$ representing a configuration over N features. The observation state s_t then corresponds to the combination of the state of the user s_t^u and the state of the environment s_t^e. The reward the device receives r_t for selecting a configuration $\mathbf{x}$ is the sum of the rewards deemed appropriate for the corresponding states of the user and the environment: $r_t = r_t^u + r_t^e$. Figure 2 illustrates the problem of AUIs as an RL task.

Following the classic RL formulation, at each step the approximation of the optimal Q-value function Q^* will be refined by enforcing the Bellman equation [Watkins and Dayan 1992], which can be reformulated for our AUI setting by substituting an action a with the task of selecting a configuration $\mathbf{x}$ given by

$$Q^*(s, \mathbf{x}) = \mathbb{E}_{s' \sim \varepsilon} \left[r + \gamma \max_{\mathbf{x}'} Q^*(s', \mathbf{x}') | s, \mathbf{x} \right], \qquad (3)$$

which states that given any state-configuration pair s and $\mathbf{x}$, the maximum cumulative reward achieved is the sum of the reward for that pair, r, plus the value of the next state we end up with, s'. The value at state s' will be the maximum over all possible configurations $\mathbf{x}'$ at $Q^*(s', \mathbf{x}')$. Thus, the optimal policy π^* corresponds to selecting the best UI configuration in any state, as specified by Q^*.

In this iterative process, the Bellman equation is used as a value-iteration algorithm to refine Q^*:

$$Q_{i+1}(s, \mathbf{x}) = \mathbb{E} \left[r + \gamma \max_{x'} Q_i(s', \mathbf{x}') | s, \mathbf{x} \right], \qquad (4)$$

where Q_i converges to Q^* as i approaches to infinity.

In the context of AUIs, we are interested in finding an optimal policy on which the device operates in order to select the best possible configuration given the states of the user and the environment. Since users in such settings experience cognitive and physical workload, they respond differently depending on: individual skills, UI familiarity, preferences, etc. In particular, these user states may correspond to implicit interactions which are often hard to detect and analyse. Nevertheless, considerable advances have been made in inferring emotional, cognitive, and behavioural states through response monitoring of various biosignals [Dinh et al. 2020], for example electrocardiography [Cairns et al. 2016], electroencephalography [Höller et al. 2017], or eye-gaze [Menges et al. 2019] and mouse [Brückner et al. 2021] movements. Thus,

such biosignals can be leveraged as an underlining technique of our approach to iteratively infer user states to find the best personalised UI configuration through an optimal policy given the inferred user states.

4 USE CASE: LEARNING MANAGEMENT SYSTEMS

With the COVID-19 pandemic, the use of learning management systems has increased dramatically [Szopiński and Bachnik 2022]. Since these systems can be accessed with different devices (e.g. desktop, mobile, tablet) and in various environments (e.g. home-office, conference rooms, common areas in universities), it is important that the UI can consider these environments for adaptation. For the purpose of demonstrating our proposed framework, we will consider a use case where an AUI accounts for changes in environment in order to improve the user's remote learning experience.

Mark is a student participating in a remote lecture that is broadcasted live on an online learning platform. Mark is attending from his apartment (environment(e)), which he shares with other students. When the lecture starts, Mark joins from his living room using a laptop and listens to the audio via laptop speakers, as he does not have any headset available. The current layout of the content presentation on his laptop (device (d)) corresponds to a configuration ($\mathbf{x}$) chosen by the AUI. Some time later, a persistent drilling noise starts coming out from his neighbour's flat, so the AUI automatically activates closed captions. The captions appear to be too small to read, which makes Mark lean closer to the screen. The AUI detects this movement and modifies the font size to improve readability. After 20 minutes, the drilling noise stops and the QA session starts, so Mark switches on his microphone to ask a question. As he begins to speak, suddenly one of his flatmates enters the living room. The AUI recognises a visual change in the environment and automatically blurs Mark's background to minimise the distracting impact that it may have on other students who are currently attending the lecture. Shortly after the doorbell rings. Mark is now alone in the apartment so he should answer the door. However, since the QA session is still going on, Mark continues listening to it via his mobile phone as he walks towards the door. The AUI modifies the layout to account for the limitations of the small screen display. The lecture concludes as Mark opens the door: it is a deliveryman with the headphones that he recently ordered.

In the presented scenario the changes in the environment such as the drilling noise (acoustic signal), someone entering the room (background visual signal), walking towards the door (posture change), etc. are signals collected from the user as well as the environment that serve as input to interpret the state of user (s_t^u) and environment (s_t^e). In RL, rewards guide the exploratory nature of an agent. Hence, in our proposed framework, the device expects some form of feedback from the user and the environment in the form of a reward (r), either positive or negative, for choosing the content presentation ($\mathbf{x}$) given their observed states. It should be noted that modelling such a reward is rather a challenging task due to difficulties in interpreting the true meaning of multi-modal signals collected from the users and their environment. Nonetheless, the above use case illustrates how modelling and responding to the changes in user environment can contribute to an improved
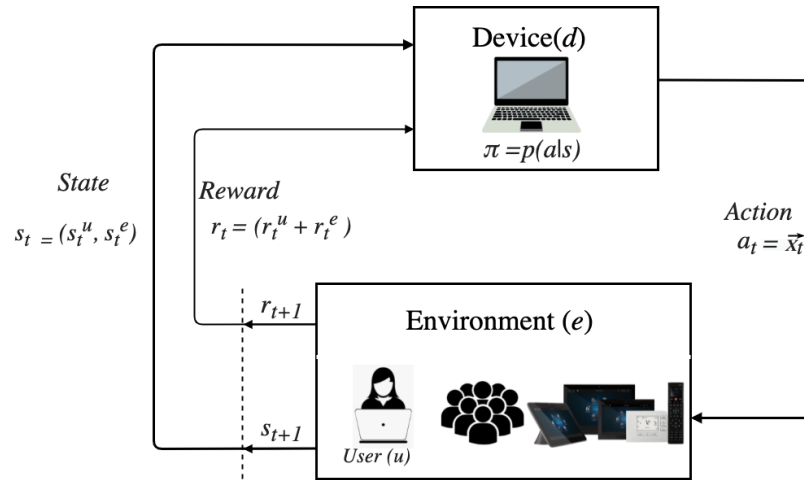
Figure 2: User interface adaptation as an RL task.

learning experience by using signals capture from the environment to automatically adapt user interface to different interaction circumstances.

5 CURRENT CHALLENGES AND IMPLICATIONS

Although our framework paves a path towards incorporating the environment in the design of AUIs, the RL formulation suffers from a scalability problem. This is due to the fact that one must compute $Q(s, \mathbf{x})$ for every state-configuration pair in order to select the best UI configuration. This is computationally infeasible as the configuration space is potentially infinite, only limited by the number of elements that can be adapted. Take for example the domain of web applications: with CSS and JavaScript it is possible to modify any UI element at will [Leiva 2011].

Recent research in RL has addressed the scalability problem by using function approximators, typically a neural network, to estimate the action-value function $Q(s, a; \theta) \approx Q^*(s, a)$ where θ is the trainable parameters (weights) of the neural network [Mnih and Kavukcuoglu 2017]. Deep Q-learning is one of the most commonly used techniques to approximate optimal action-value functions using a neural network. Hence, we can define our Q-function approximator using a neural network too. This means that, in the forward pass while training the network, we use a loss function to minimise the error of the Bellman equation, thereby determining how far $Q(s, \mathbf{x})$ is from the target $Q^*(s, \mathbf{x})$, given by

$$L_i(\theta_i) = \mathbb{E}_{s, \mathbf{x} \sim \rho(.)} \left[y_i - Q(s, \mathbf{x}; \theta_i) \right]^2 \tag{5}$$

where, $y_i = \mathbb{E}_{s' \sim \varepsilon} \left[r + \gamma \max_{\mathbf{x}'} Q(s', \mathbf{x}'; \theta_{i-1}) | s, \mathbf{x} \right]$. Then, then backward pass is a gradient update with respect to the Q-function parameters θ.

It is also evident from recent works that there are different variations of DQN that have enjoyed a huge success in RL tasks such as Actor-Critic methods [Xiang et al. 2019], which combine DQN with Deep Deterministic Policy-Gradient Algorithms (DDPG) [Lillicrap et al. 2015] and a multi-agent version of actor-critic methods [Lowe et al. 2017; Ryu et al. 2020]. We are optimistic that our proposed

framework, complemented by such techniques, may overcome scalability issues and open new, interesting research opportunities in AUIs, as it enables adaptation by learning through exploration.

Overall, our proposed contextual framework can bring several practical benefits. Firstly, it allows a contextual modelling approach of AUIs that incorporates environment and accounts for its constituent elements. Secondly, framing the problem as an RL task opens possibility of multidisciplinary research at the intersection of Machine Learning, Cognitive Science and Psychology. Thirdly, if applied in practise, the RL architecture empowers us to learn complex interaction responses of users and their environment through exploration, which, in turn, can lead to design of more seamless and user-friendly AUIs. Furthermore, although we have used a single user setting to illustrate the proposed architecture, it could be extended to more complicated interaction scenarios such as multi-user setting and group adaptation, where single adaptation is applied to multiple users.

Nevertheless, we would like to note that there are several open questions that need to be addressed before the practical implementation of our framework becomes feasible. Firstly, determining the reward function poses a challenge as it requires keeping the user in the loop in order to provide their feedback based on system's actions. Secondly, the system needs to correctly interpret and act on a multitude of signals from the environment that may potentially have contradictory meaning. Thirdly, due to complexity of the environment, the neural network requires to be exposed to a large number of examples which makes training extremely time consuming and negatively impacts scalability of the system.

We anticipate that by concerted collaborative efforts, the AUI community will begin to address these challenges to pave way to practical implementation of the proposed contextual framework in the future.

REFERENCES

Silvia Abrahão, Emilio Insfran, Arthur Sluÿters, and Jean Vanderdonckt. 2021. Model-based intelligent user interface adaptation: challenges and future directions. *Softw. Syst. Model.* 20, 5 (2021).

Kai Arulkumaran, Antoine Cully, and Julian Togelius. 2019. AlphaStar: An Evolutionary Computation Perspective. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO)*.

Sarah Bouzit, Gaëlle Calvary, Joëlle Coutaz, Denis Chêne, Eric Petit, and Jean Vanderdonckt. 2017. The PDA-LPA design space for user interface adaptation. In *Proceedings of International Conference on Research Challenges in Information Science (RCIS)*.

Lukas Brückner, Ioannis Arapakis, and Luis A. Leiva. 2021. When Choice Happens: A Systematic Examination of Mouse Movement Length for Decision Making in Web Search. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*.

Andrew W. Cairns, Raymond R. Bond, Dewar D. Finlay, Cathal Breen, Daniel Guldenring, Robert Gaffney, Anthony G. Gallagher, Aaron J. Peace, and Pat Henn. 2016. A computer-human interaction model to improve the diagnostic accuracy and clinical decision-making during 12-lead electrocardiogram interpretation. *J. Biomed. Inform.* 64 (2016).

Jim X Chen. 2016. The evolution of computing: AlphaGo. *Comput. Sci. Eng.* 18, 4 (2016).

Toan Dinh, Thanh Nguyen, Hoang-Phuong Phan, Nam-Trung Nguyen, Dzung Viet Dao, and John Bell. 2020. Stretchable respiration sensors: Advanced designs and multifunctional platforms for wearable physiological monitoring. *Biosens. Bioelectron.* (2020).

Kunihiko Fukushima and Sei Miyake. 1982. Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets*. Springer.

Krzysztof Gajos and Daniel S Weld. 2004. SUPPLE: automatically generating user interfaces. In *Proceedings of the International conference on Intelligent User Interfaces (IUI)*.

Yvonne Höller, Andreas Uhl, Arne Bathke, Aljoscha Thomschewski, Kevin Butz, Raffaele Nardone, Jürgen Fell, and Eugen Trinka. 2017. Reliability of EEG Measures of Interaction: A Paradigm Shift Is Needed to Fight the Reproducibility Crisis. *Front. Hum. Neurosci.* 11 (2017).

Yue Jiang, Ruofei Du, Christof Lutteroth, and Wolfgang Stuerzlinger. 2019. ORC layout: Adaptive GUI layout with OR-constraints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

Luis Leiva. 2012. Automatic web design refinements based on collective user behavior. In *Proceedings of CHI Extended Abstracts on Human Factors in Computing Systems*.

Luis A. Leiva. 2011. Restyling Website Design via Touch-Based Interactions. In *Proceedings of the International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI)*.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning.

Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in neural information processing systems (NeurIPS)*, Vol. 30.

Mark W. Maier. 1996. Architecting Principles for Systems-of-Systems. In *INCOSE International Symposium*, Vol. 6.

Raphael Menges, Chandan Kumar, and Steffen Staab. 2019. Improving User Experience of Eye Tracking-Based Interaction: Introspecting and Adapting Interfaces. *ACM Trans. Comput. Hum. Interact.* 26, 6 (2019).

Volodymyr Mnih and Koray Kavukcuoglu. 2017. Methods and apparatus for reinforcement learning. US Patent 9,679,258.

Thanh Thi Nguyen, Ngoc Duy Nguyen, and Saeid Nahavandi. 2020. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Trans. Cybern.* (2020).

Heechang Ryu, Hayong Shin, and Jinkyoo Park. 2020. Multi-agent actor-critic with hierarchical graph attention network. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Vol. 34.

Thomas Serre, Lior Wolf, and Tomaso Poggio. 2005. Object recognition with features inspired by visual cortex. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 2.

Anil Shankar, Sushil J Louis, Sergiu Dascalu, Linda J Hayes, and Ramona Houmanfar. 2007. User-context for adaptive user interfaces. In *Proceedings of the International conference on Intelligent User Interfaces (IUI)*.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, and Marc Lanctot. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 7587 (2016).

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, and Adrian Bolton. 2017. Mastering the game of go without human knowledge. *Nature* 550, 7676 (2017).

Ke Sun, Chun Yu, Weinan Shi, Lan Liu, and Yuanchun Shi. 2018. Lip-interact: Improving mobile device interaction with silent speech commands. In *Proceedings of the Annual ACM Symposium on User Interface Software and Technology (UIST)*.

Richard S. Sutton and Andrew G. Barto. 1998. *Introduction to reinforcement learning*. Vol. 135. The MIT Press.

Amanda Swearngin, Chenglong Wang, Alannah Oleson, James Fogarty, and Amy J Ko. 2020. Scout: Rapid exploration of interface layout alternatives through high-level design constraints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

Tomasz Szopiński and Katarzyna Bachnik. 2022. Student evaluation of online learning during the COVID-19 pandemic. *Technol. Forecast. Soc. Change* 174 (2022).

Kashyap Todi, Gilles Bailly, Luis Leiva, and Antti Oulasvirta. 2021. Adapting User Interfaces with Model-Based Reinforcement Learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

Yuntao Wang, Jianyu Zhou, Hanchuan Li, Tengxiang Zhang, Minxuan Gao, Zhuolin Cheng, Chun Yu, Shwetak Patel, and Yuanchun Shi. 2019. Flextouch: Enabling large-scale interaction sensing beyond touchscreens using flexible and conductive materials. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019).

Christopher JCH Watkins and Peter Dayan. 1992. Q-learning. *Mach. Learn.* 8, 3-4 (1992).

Jiaqi Xiang, Qingdong Li, Xiwang Dong, and Zhang Ren. 2019. Continuous control with deep reinforcement learning for mobile robot navigation. In *Chinese Automation Congress (CAC)*.

Yukang Yan, Chun Yu, Xin Yi, and Yuanchun Shi. 2018. Headgesture: hands-free input approach leveraging head movements for hmd devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018).