VirtuAUL - A Design Framework for Adaptive Lighting

Toni Österlund\textsuperscript{1}, Henrika Pihlajaniemi\textsuperscript{2} \\
\textsuperscript{1,2}University of Oulu \\
\textsuperscript{1,2}\{toni.osterlund\|henrika.pihlajaniemi\}@oulu.fi

The rapid development of lighting technology and the integration of information technology into our everyday environments, have opened up new possibilities for the creative and sustainable design of adaptive lighting. For the design, control and real-world implementation of adaptive lighting, we present a novel design framework called VirtuAUL. The framework consists of computational methods and tools that allow for the lighting designer to design adaptive lighting processes, that generate lighting patterns based on sensor stimuli, without the need for programming. The control and design methodology is based on the deployment of network-based agents: designer controls and guides flows of agents in a virtually defined network, where lights and sensors act as the network nodes. The VirtuAUL designer software allows for the designer to graphically define the adaptive processes that generate the lighting patterns by modifying the network nodes, topology, link directionalities, agent parameters and other design elements.

Keywords: Adaptive lighting, Design framework, Lighting design, Network-based agents, Design tool

INTRODUCTION

The rapid development of lighting technology, especially the introduction of LED, and the integration of information technology and infrastructure into our everyday environments, have opened up new and exciting opportunities for lighting design. Intelligent control systems, combined with the use of different sensing technologies, allow us to use interactive and adaptive processes as an integral part of the lighting scheme.

So far, the introduction of adaptive lighting solutions in indoor and urban spaces have been mainly motivated by efforts to conserve energy through automation and optimised solutions (see for instance Hughes & Dhannu 2008). However, although technologically sophisticated, the automated and optimised systems do not consider the role of design and aesthetics in the lighting solutions (Röcker et al. 2012). While energy conservation is an important task of future lighting systems, we want to open up new possibilities for the creative and sustainable design of adaptive lighting. Adaptation in lighting can be designed to offer aesthetic experiences, communicate and convey information as well as offer opportunities for social interaction (Pihlajaniemi et al. 2013).

Adaptive lighting is seen as an intelligent system, where lighting dynamically reacts and adapts to the presence of people and current environmental conditions through environmentally sensitive informa-
The adaptive processes of lighting can entail aesthetic qualities and artistic expression through creative design process. (Figure 1)

**Designing adaptive lighting**

In order to design adaptive lighting solutions, the designer has to define the system's behaviour and responses to sensor stimuli. This is a challenging design task, as the lighting system can comprise of multitude of different light sources and sensors, with all of them having different controllable parameters. The complexity of the system and the sheer amount of sensor and other data necessitate at least some degree of computational intelligence, where the adaptive behaviour is controlled by an autonomous process coupled with a method for explicit or implicit user interaction (van Essen et al. 2012).

Designing adaptive lighting with the tools currently available requires knowledge and skills in programming, yet the creative professionals who possess expertise in designing functional and experiential lighting, may not have skills in coding the necessary algorithms. This might create a gap between the functional and aesthetic intentions of the designer and the design potential of adaptive lighting, which could lead to overly simplified and linear design solutions mainly concerned aspects of energy saving. There is a need for new concepts, generalizable design methodology and tools for the design of adaptive lighting, as also pointed by (Karamouzi et al. 2013; Hakulinen et al. 2013A).

Currently, adaptive lighting processes are defined either by coding the algorithms directly on to the physical lighting controller unit or by constructing purpose-built design tools. These design tools can be used to simulate the results and they allow the visual exploration of the design parameters. Also commercially available lighting controllers' design software can be used to aid in the design of adaptive lighting by utilizing elements in their native user interfaces (for example LightAct Designer, Traxon e:cue and Pharos Designer). If needed, more complex behavioural logic have to be implemented by extending their functionalities via scripting, if the design
software provides this possibility. For the creation of small purpose-built design tools, several programming toolkits and integrated development environments (IDE) exist for "creative coding" that provide simplified methods for handling media and visual content, such as Processing, OpenFrameworks and Cinder (Brewis 2014).

For the creative design, control and real-world implementation of adaptive lighting, we have developed a design framework called VirtuAUL. It is not a single design software, but it refers to a framework of computational methods and solutions that allow for the lighting designer to design, simulate and implement adaptive lighting processes with ease. The framework incorporates a novel network-based agents methodology (Figure 2) for the control and design of adaptive lighting, and a graphical design tool implementing the method.

![Figure 2](Image)

The flow of agents in the virtual system emerges as dynamic and adaptive lighting patterns as the lighting system interacts with the real-world and its users. Information is conveyed from the real to virtual through sensors, and lights reveal the distribution of the virtual agents in the system.

Related work

The field of adaptive and intelligent lighting has been previously researched from multiple perspectives, such as energy consumption (Hughes & Dhannu 2008) and city safety (Haans & de Kort 2012). Interaction has been investigated through different methods for implicit and explicit interaction (Dugar et al. 2012; Offermans et al. 2014; Pihlajaniemi et al. 2014B) and in connection to specially designed hardware in adaptive lighting environments (Magielse 2014). An intelligent hybrid control approach has been used in order to examine the balance between automated and optimised lighting and the preferences of the user (Offermans et al. 2013).

The authors developed an online participatory lighting design tool, called LightStories which offered users means to define an hour long light animations to be shown on a pedestrian street by RGB-LED ribbons on street light poles. Though not real-time interactive, as the animated light patterns were triggered by a set calendar time, it allowed users to construct a design from eight different pre-defined static or dynamic lighting effects, with all varying design parameters. (Pihlajaniemi et al. 2012) Poulsen et al. (2013) created a simple design software for their responsive public lighting installation. A version of the tool was also created for the Android mobile operating system, which allowed for the city users to select a pre-set behaviour and modify its parameters, and to simulate the outcome. Hakulinen et al. (2013B) created story-based exercise games for children, where interactive lights guide and motivate children. The games are modelled as state machines using XML markup and states are changed by an operator using a controller.

OBJECTIVES AND METHODS

The development of the VirtuAUL design framework stemmed from the need to solve specific design and implementation challenges concerning two real-world adaptive lighting demos: 1) Urban Echoes (UE): adaptive and communicative lighting in an urban park environment, and 2) intelligent and Adaptive Re-
The objective was to research and develop new methods and tools for the creative and functional design of adaptive lighting. The needs and goals of the development were clarified during the design and construction processes of these lighting demos.

In addition to the new design methodology, the development aimed to bring new aesthetic qualities of unpredictable factors and changes into adaptive lighting, in contrast to the linear and predictable optimisation processes. The qualities are present in the beauty and play of natural elements, such as changes in weather, light and wind or the swarming behaviour of birds and fish. Adding a layer of unpredictable, nonlinear dynamic changes into lighting, in addition to rationally and logically linear changes, would provide a novel aesthetic and meaningful approach to lighting design and the design of lighting as ambient media of architectural spaces.

**Real-world adaptive lighting demos**

UE was a temporary park lighting installation at Otto Karhi park in the centre of Oulu, a city in northern Finland. The lighting installation was a flexible system, which could produce both even distributions of light and more uneven distributions consisting of different sized patches of light on the park pathway. The lighting reacted to park visitor's movements via IR-motion sensors, pursuant to different lighting scenarios ranging from monochromatic slow-paced and linear, to multi-coloured fast-paced nonlinear adaptive processes. The purpose of the demo was, besides developing the design framework, to study different adaptive lighting scenarios from the perspective of a park visitor's experience, as well as communicative potential of urban lighting.

The main methodologies and functionalities of the design framework were developed iteratively during the design process of this demo. The design of the UE demo laid out the basic needs for the design method and tool, and functionalities were added based on the need of different preconceived adaptive lighting scenarios, and how to enable or to ease their design. The description of the communicative features and the transdisciplinary development process are discussed in detail in (Pihlajaniemi et al. 2014C) with an evaluation of results.

ARL was a lighting pilot at the ladies' clothing section, where adaptive lighting was designed to enhance the shopping experience, as well as to influence the customers' shopping behaviour. The pilot environment was a part of a ladies' clothes section with controllable neutral white LED luminaires for general lighting and warm white LED spotlights and RGB-LED spotlights for target lighting and colour effects. Three different adaptive scenarios were designed: "Static White", "Adaptive White" and "Adaptive Colour" scenarios. The design of the scenarios followed a three-mode strategy in response to the customers' movements in the test area. The adaptation modes were "Attract", "Focus" and "Keep in the area". (Pihlajaniemi et al. 2014A)

In the demo, the customers' positions and movements were tracked via network of depth cameras. These provided detailed coordinate and vector information, as well as a pre-processed categorisation of the person based on their position, direction and behaviour, which related to the three modes of adaptation. The possibility to use coordinate data and type categorisation created new sensing, design and control possibilities that were implemented to the design tool, as well as some general improvements to the tool's functionalities.

**VIRTUAUL DESIGN FRAMEWORK**

The VirtuAUL design framework can be used in lighting systems of different sizes, effectively scaling up to an urban block or a city district. The real-world processing happens in a centralised controller unit, which receives information about the environment through sensors and other data sources. The use of a centralised controller unit limits the scalability of the system compared to decentralized systems, in which the processing and data transfer is distributed among multiple intelligent hardware components (Rao et al. 2012). However, using a centralised con-
controller makes it possible to utilize 'unintelligent' off-the-shelf hardware and no special components are necessarily needed.

The VirtuAUL design framework (Figure 3) consists of four parts that together define the control, design and implementation processes of adaptive lighting from the designer to real-world lighting: 1) the base control methodology of network-based agents that defines the control process and aspects of the design elements, 2) VirtuAUL designer software for designing and simulating the adaptive control processes, 3) XML-based save file and a container for the designed lighting scenario that can be transferred to the central controller unit, and 4) VirtuAUL controller software that runs in the lighting controller unit and operates the lighting in real-world. The next section focuses on the functionality of the design tool and a brief overview of the control methodology is given here below for a better understanding of the framework as a whole.

**Network-based agents**

The network-based agents methodology define a process, in which multitude of autonomous information conveyor, or agents, move along the links of a designer-defined virtual network, where the nodes represent physical light fixtures and sensors. When sensor information reaches a defined threshold (i.e. sensor gets triggered), the sensor node start to emit agents to the network. Light nodes react to the vicinity and properties of the agents by affecting the colour and intensity of light of the corresponding physical light fixtures. (Figure 4)

The network-based agents control method is inherently dynamic and it creates ever-changing lighting patterns that are the complex sum of the behaviour and properties of the multiple agents and the network. The methodology brings forth three main elements that together form the main parameters of the control process: virtual agents, network nodes and network links. These become the main design ele-
ments and ways to affect the process of the adaptive lighting scenarios.

The network-based agents system is in its essence a multi-layered multi-agent situated system (MMASS) (Bandini et al. 2002), in which the environment where the agents live is explicitly structured and spatially related to the real-world environment, which makes the system situated (Ferber & Müller 1996). Situatedness can also be defined as that the agents receive sensory input from their environment and perform actions which change that environment in some way (Jennings et al. 1998). Multi-layered (Bandini et al. 2009) in this context, refers to the fact that differentiated nodes, links and agents exist within the same system, allowing multiple and completely separated networks with their distinct agent types to exist within the same virtual system. This means that one adaptive lighting scenario can define and simultaneously maintain very different kinds of behaviours of light. The network-based agents system utilises graph-based coordinates as the spatial model, where the hierarchy and structure of the lights to themselves and to the environment are defined by the topology of the network (Österlund 2013).

Although the concept of an agent is not new and is widely used, there is currently no precise definition nor an universal agreement of what an agent is (Macal & North 2009; Jennings et al. 1998). For the purpose of this research, an agent is considered as a self-contained computational component that is capable of autonomous behaviour, situated in a virtual environment and capable of interacting with the real world through sensor information and lighting. Agents convey and transmit their parameter information to the lighting system. They are memoryless, reactive (Ferber & Müller 1996) and situated (Weyns & Holvoet 2004) with defined dynamic positions, directions and velocities.

VIRTUAUL DESIGN TOOL

The VirtuAUL designer is a design tool that implements the network-based agents methodology, and allows the designer to graphically define the parameters and configurations of the different design elements without the need for programming. The first versions of the tool were implemented in Java using the Processing IDE, and were later ported to the Eclipse IDE.

The VirtuAUL designer (Figure 5) is divided into three separate work areas: 1) In the middle is the Design View (DV) that displays the network, its nodes and links, and visualises the agent flows. Other rel-
evant design parameters can be accessed and modified through menus. 2) In the right is the Simulation View (SV), where the designed lighting scenario can be simulated on top of a two-dimensional vector map of the design area. The lights and sensors are represented on the map in SV, and the corresponding nodes are shown in DV. 3) In the left there are additional Control Elements that can be used to operate some functionalities as well as set visibilities of GUI elements.

The design process and implementation of adaptive lighting with the VirtuAUL designer includes four basic phases: 1) set-up, 2) design, 3) simulation and 4) implementation phases. Each of them consists of several actions and they can be executed iteratively. (Figure 6)

Set-up. The set-up phase involves the modelling of the physical lighting infrastructure, i.e. the light fixtures and sensors, their locations and properties, into the tool. The light fixtures and sensors are placed on top of the map in SV, where they are represented as symbols with orientation, size, and lighting and sensing areas. The properties of the light fixtures and sensors inside the tool have to be visually set to match their real-world counterparts. Especially sensing areas and orientation have to be accurately set, or otherwise the simulation and the real-world implementation differ.

Nodes are identified by their unique ID-numbers, which correspond to the path addresses of the physical light fixtures connected to the system. The control information from VirtuAUL to the lights is conveyed solely based on the IDs and path addresses, so their correct set-up is important. The set-up configuration can be saved as a template file, still void of any design information. The template is a basic save file, but as the physical infrastructure is unlikely to change, it can be used as a quick starting point in the design of different lighting scenarios.

Design. After the system infrastructure is modelled into the tool, the design commences by defining the network topology. As sensor nodes emit the agents into the system, a direct link from a sensor node to a light node defines a lighting reaction to sensor stimuli. Regardless of the physical distance between linked light and sensor nodes, the agent's speed defines the time it takes to pass through a link. As the actual real-world distance is not considered, sensors can trigger simultaneous effects at any part of the overall lighting scene. By defining the links, designer effectively defines the pathways for the flows of light.

The light patterns' properties, such as colour, transition speed and duration are controlled mainly through modifying the agents' parameters. An individual agent does not have a significant influence on the resulting lighting, but the lighting patterns emerge from the dynamics of multitudes individual agents and their combined properties.

Agents that share parameters are described as a Family, and share family traits without being completely identical. Fluctuation in some parameters create variation to the movements of individual agents, as otherwise a group of identical agents would move in unison, indistinguishable from a single agent. Different families can be defined using a graphical controller and their properties define different behaviour. For instance, one family could consists of brightly coloured agents that move fast, but die young, and another consisting of white agents that move slow and live longer. When emitted to the system, their properties emerge as very different patterns of light.

Together with the light and sensor nodes, there are also two other node types; ghost nodes and a control algorithm nodes. The ghost node is an empty node that extends the agent pathway. The control algorithm node determines the sensor trigger and threshold, and the type and amount of agents that are emitted onto the system. Sensor nodes themselves can represent physical sensors, such as IR-motion sensors in the UE demo, or virtual sensors as
in ARL. Virtual sensors sense and react to externally input information, such as the location of people in the area. (Figure 7)

Using the control node, a designer can determine the amount (generated per timeframe), family and link layer of the emitted agents. When connected to a virtual sensor the designer can select, which user categorisation the sensor reacts to, and by using a radar chart controller, which movement directions are taken into account. This allows, for instance, to activate certain lighting situations only for users that are heading towards the lighting effect in a pace that they have enough time to see and react to it. Different control nodes with different configurations can be connected to a single sensor, which can simultaneously trigger multiple lighting effects.

The agents can be emitted on to specific link layers. The use of link layers allows for the designer to define link directionalities or close access, thus creating different topological configurations for each layer. The directionality of a link determines, whether a link is bidirectional, unidirectional (with direction control), or closed. If an agent does not have a layer specified, all links are considered as bidirectional for that agent.

**Simulation.** The creation of the network, its topology and the characteristics of the agents have a vast influence on the colours and patterns of the light and the overall speed of changes. As the control method is inherently nonlinear in nature, small changes in, for example, agent parameters can have big implications in the overall lighting patterns. This makes the real-time simulation of the system important. The simulation does not thrive for physical accuracy, but it visualises the appearance of the dynamic adaptive process and resulting lighting behaviour. Without visual feedback on the effects of the design actions and modifications made, it would be challenging for the designer to evaluate the design outcome. The information of the simulation run can be stored as a CSV file, containing the colour information of each light, which can then be analysed according to their energy usage.

The simulation of the designed lighting scenario happens by moving the pointing device on top of the map in SV. The sensors located in the area register the movement and behave according to their configurations. The lights in SV are displayed as pre-rendered light distribution images. The dynamics of the adaptive lighting is simulated by constantly changing the tint and or transparency of the light image, which creates the appearance of a dynamically changing...
light. Simulation is visualised in DV as moving agents, shown as coloured spheres, and it simultaneously reveals the inner process of the system.

Implementation. The designed adaptive scenario is a dynamic process, and a method describing it was developed as an XML-based file that hierarchically explains the network connections, agent characteristics and design parameters. The file does not contain programming code, but only stores the design information so it can be reproduced in the design tool or with the VirtuAUL controller software running on a lighting controller.

The VirtuAUL controller is otherwise identical with the VirtuAUL designer, but it lacks all the elements of the GUI and the possibility to modify the information. The process is not controller specific and can be used in commercial or custom built controllers as long as they are capable of running and maintaining a custom program and read the contents of the design file. All changes and new functionalities that are implemented in the design tool must be matched in the controller software. The VirtuAUL controller was written with the LUA scripting language.

The VirtuAUL controller parses the design file and sets up the virtual network by matching node IDs with light fixtures' DMX addresses and physical sensors. The system state and agent positions are updated several times per second and the overall control data to the affected lights is generated. After each timeframe the control data is gathered and conveyed to the lights.

DISCUSSION
In this paper, we have presented the VirtuAUL design framework implementing the network-based agents methodology for controlling and designing adaptive lighting scenarios. The design of the scenarios is done using the graphical software tool, called VirtuAUL designer, where the designer visually defines and explores the design element parameters, and can simulate the results for a desired result. The benefit of the method and design tool is in the visual definition and evaluation of the design decisions using real-time simulation and visualisation of the process. It allows to define a wide range of different types of lighting behaviour as flows of light, rather than hard to describe and hard to control coding of algorithmic processes.

The behaviour of the lighting can be creatively designed, ranging from predictable linear solutions to unpredictable nonlinear solutions. The nonlinearity of the lighting behaviour is affected by the number of agents emitted and the number of possible routes they can take, as well as the use of different link layers and agent types (based on families). The specially implemented constrictions, such as the link layers, link directionality and agent path selection behaviour, ease the explicit control of the flows of agents, making it easier to define linear lighting solutions.

The possibility to view the visualisation of the flows, side-by-side with the resulting simulation, simplifies the design exploration. The simulation happens in 2D by visualising light effects on top of a map image. The creation of a static "light map" image is commonly used visualisation and design method in lighting design, and the implementation of 2D simulation to the tool is simpler than the creation of a 3D simulation environment. However, the real-time simulation of lighting effects in an accurate 3D environment would be an interesting path for further development.

REFERENCES
Brevis, D 2014 'Emerging Environments for Interactive Art Creation', Interactive Multimedia Conference 2014, University of Southampton
Dugar, AM, Donn, MR and Marshall, S 2012, 'Designing
Tangible Lighting Control Interfaces', *LEUKOS*, 8(3), pp. 215-228

van Essen, H, Offermans, S and Eggen, B 2012 'Exploring the Role of Autonomous System Behavior in Lighting Control', *DIS 2012, workshop on Designing Interactive Lighting*, NewCastle, UK


Haans, A and de Kort, YA 2012, 'Light distribution in dynamic street lighting: Two experimental studies on its effects on perceived safety, prospect, concealment, and escape', *Journal of Environmental Psychology*, 32(4), pp. 342-352


Offermans, S, van Essen, H and Eggen, B 2013 'Exploring a hybrid control approach for enhanced user experience of interactive lighting', *Proceedings of the 27th International BCS Human Computer Interaction Conference*


Weyns, D and Holvoet, T 2004, 'A formal model for situated multi-agent systems', *Fundamenta Informaticae*, 63(2), pp. 125-158