A Connected Swarm Cycling System

1st Linglong Meng, 2nd Stefan Schaffer, 3rd Vincent Wappenschmitt
Cognitive Assistants
German Research Center for Artificial Intelligence (DFKI)
Berlin, Germany
linglong.meng@dfki.de, stefan.schaffer@dfki.de, vincent.wappenschmitt@dfki.de

***Abstract*—Social group cycling shows a positive impact on facilitating urban cycling as a sustainable means of mobility while increasing cycling safety in urban areas [1]. We present a new urban mobility concept, *Connected Swarm Cycling*, that creates a group of people cycling together for a while in a common direction or destination. We assume that the concept of Swarm Cycling can significantly change the mobility behaviour of citizens and will be a building block of green mobility for sustainable cities in the future. Utilizing an OSRM [1] routing service with support of trip intersection computing, the system inducts the cyclists into a cycling swarm. The swarms are formed automatically via peer-to-peer connection when cyclists come in proximity, and the information of the swarm and individual cyclist will be synchronized within the swarm via a Nearby Mesh Network. Supporting the implicit interaction within or between swarms, smart wearables are utilized to realize use cases like swarm member identification or signalling in case of merging or splitting of swarms. In this paper, we also present a technical description of our system, including the protocol and network model to support the coordination and synchronization within the swarms.**

***Index Terms*—swarm cycling, sustainable mobility, implicit interaction**

I. INTRODUCTION

Recent research shows that in 2019 cycling accounts about 8% of all passenger trips in the EU [2]. Increasing the number of citizens using bicycles for everyday mobility discloses huge potentials to reduce congestion, define holistic transport concepts, reduce commuting time, and to plan and create a sustainable mobility infrastructure. However, cycling is still a challenge for the safety of urban mobility. Due to the insufficiently developed transport infrastructure, cyclists often find themselves in a situation where they have to share the path with pedestrians or motor vehicles. Since road infrastructure is often designed to prioritize the efficient movement of vehicles, such as cars and trucks, lack of presence while cycling alone put the cyclists even in more dangerous situation where traffic is intensive. Expressed in numbers for the city of Berlin, 5 cyclists were killed in 2018, 6 in 2019, and already 10 in July 2020[2]. Additionally, cyclists in urban areas often suffer from problems such as bad infrastructure, or aggressive behaviour of motorised road users.

In order to improve the situation of urban cycling, we introduce a new form of urban mobility, namely connected swarm cycling. The ”safety in numbers” approach states that when more people are walking or biking, the likelihood of a collision with a pedestrian or bicyclist is lower [3]. Cycling in a group enables cyclists to gain better visibility on the road as they ride as one bigger visible unit. Current state of the art already experiments with automatically inducting cyclists into a collective of nearby riders, letting their bicycle lights begin to pulsate in unison if cyclists get close to each other [4]. According to the authors, this creates a visually unified presence and extending membership in a self-organizing community that is increasing safety cooperatively. Wireless communication technology like WiFi, Bluetooth and ZigBee enable the exchange of information among bicycles with an ad-hoc connection [9]. Céspedes et al. introduced a platoon-based cyclist cooperative system that provides safe and efficient coordination for groups of cyclists [8]. Coordinated groups of cyclists can also have a significant impact on the green wave experience in urban mobility. In Germany groups of cyclists with more than 15 participants are allowed to ride together as a unit in road traffic. Cycle paths that must be used by solo cyclists are of no significance for such groups[3]. Besides, naturally people in the urban environment tend to congregate in social groups and often develop a sense of social solidarity. We expect that both citizens who already bicycle and others who do not yet bicycle often will be attracted to the benefits of participating in bicycle swarms. Overall, the implementation of the concept will lead to an increase in bicycle use in the city and thus to a change in the mobility behaviour of the users of the cities' mobility system. The concept also supports the transition to a more green and sustainable mobility service to be managed by local authorities.

In this paper we present the design and implementation of a swarm cycling system that dynamically forms groups of cyclists for everyday routes from A to B in the city. We first describe the concepts and the use cases of swarm cycling. Thereafter we provide a technical description of the swarm cycling system that includes the design of system, peer-to-peer communication protocol and the topology of mesh network which focus on the formation of swarm and the changes of members within the swarm. Next, the smart wearable

---

[1] http://project-osrm.org/

[2] https://adfc-berlin.de/radverkehr/sicherheit/information-und-analyse/145-unfallorte.html

[3] https://de.wikipedia.org/wiki/Verbandregel

prototype is presented in terms of hardware setup and software implementation. Thereafter a detailed description about the different scenarios in swarm cycling is presented. At last, we conclude and highlight the future works.

II. SWARM CYCLING

The idea of connected swarm cycling is to utilize the current mobility context of a user to create a so-called “swarm”, a group of people cycling together with a common direction or destination. The general concept is based on the outcomes of the EU projects STREETLIFE and BIG IoT –BIKELIFE [5]. The technological outputs of these projects were put together into the open-source mobility app *Bikerider* which provides a routing service for bicyclist. A unique feature of *Bikerider* is that the suggested routes also show overlapping parts with other cyclists. These overlaps are defined as trip intersections. Through visualized trip intersections in the user interface, bicyclists can find potential swarm routes.

We define the beginning waypoint of the trip intersection as the check point for the bicyclists that share the trip intersection. Before the bicyclists arrive at the check point, they will be notified about the information of each other such as relative distance and remaining cycling time to the check point. When the bicyclists are in proximity, a swarm is formed via peer-to-peer connection. If there are more than two bicyclists, a mesh network will be established to propagate information and synchronize within the swarm group.

Smart wearables are utilized to communicate the presence and proximity of groups. When a bicyclist is joined by another bicyclist, the smart wearables will notify both bicyclists. As the swarm is synchronized from time to time, if any bicyclist leaves the proximity of others, the swarm will sense that and notify the bicyclist and the rest of swarm members as well. To facilitate the creation of dynamic mobility swarms, the smart wearable of individual user is able to memorize the current group membership and to exchange information with other smart garment in range directly or via smartphone. As the presence of swarm group grows with the numbers, the safety of the bicyclists is increased. Additionally, as the size of a swarm grows, the mesh network expands as well, thus enhancing the potential growth of the swarm.

Another feature of the swarm cycling system is that interactions of the user within the swarm group are conducted implicitly, which provides the advantage that the users can focus on their current actions (i.e., the cycling task) and do not have to be aware of the system or are distracted by the system [6]. Implicit interaction does not require explicit commands from a user. Instead, the activity of the user is interpreted as input in the specific context, whereas the system accordingly initiates appropriate actions. In swarm cycling, the situational contexts of the user such as the location (GPS coordinates) or the state of devices (e.g., peer-to-peer connection state) are utilized as implicit input. The system will generate the appropriate actions by using geofencing and the peer-to-peer connection detection for different use cases, e.g., arrive a check point, join a swarm, etc. For example, when a new swarm member is approaching the swarm, a notification via the swart wearables can be perceived peripheral with very low distraction (e.g., through vibration in a pulsing mode), so that the cyclists in the swarm group could still pay attention to the current traffic situation while the group can prepare for the newly joining member.

III. SYSTEM DESCRIPTION

A. Overview of the swarm cycling system

The swarm cycling system is based on client-server architecture. Figure 1 illustrates the high-level architecture of the swarm cycling system. The front-end of the system comprises of an Android application *Bikerider* and smart wearables which are connected with the smartphone via Bluetooth. *Bikerider* provides the cyclists with a multimodal input interface for trip request. The cyclist is able to input the start point and the destination point per voice or text, or GUI (by pointing on the map interface) as well. The response of a successful trip request consists of the trip recommendations with information of trip intersections with the other bicyclists, which provides the cyclists the option to select the preferred route. The Nearby Mesh Network module integrated into the application is there for synchronization and information propagation within the swarm group. The user interface also consists of smart wearables which are used for signalling in different use cases, e.g. join a swarm, leave a swarm, etc.

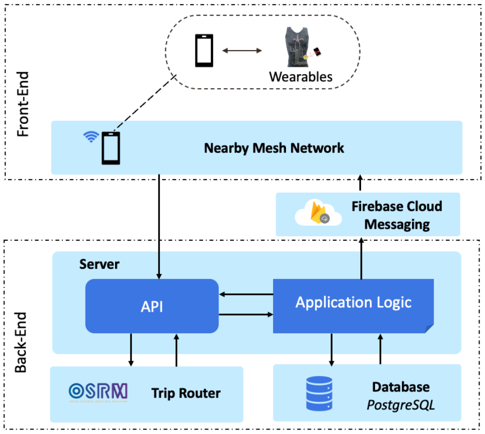


Fig. 1. High-level system architecture of Swarm Cycling

On the server side, the routing service with trip intersection computing is provided by the back end utilizing the Open-Source Routing Machine (OSRM). The server processes the trip requests from the cyclist sent via a REST API and further redirects the processed requests to a running instance of the OSRM with a loaded bicycle profile. Thereafter, the application logic module will compare the trip proposals from OSRM with ongoing trips of other cyclists in the database to determine the trip intersections. The trip proposals with trip intersections will be sent back to the front-end as response.

After the cyclist selects a route, the front-end application regularly shares the current position of the cyclists with the server so that the server can broadcast location-aware notifications to individual cyclists via Firebase cloud message service. Besides that, the Firebase cloud message service is also used to synchronize the trip intersection information if there is update with the trip intersection, for example, a new cyclist starts a trip that share the trip intersection with the current cyclists. For the moment the server also serves as a database server for the swarm data. The data could be further used for the functionalities such as showing nearby swarm in real time, coordinating different swarms, etc., but are left for future development.

B. Nearby Mesh Network

In Swarm Cycling, the coordination within the cycling swarms is designed to be implicitly without the requirement of an explicit interaction from the cyclists. Therefore, the peer-to-peer connection between cyclists should be completely automated, more specifically, peer discovery, peer advertising and connecting should be done automatically without user interaction. We use the Google Nearby Framework[4] which supports device-to-device exchanges by implementing network advertisement, discover and communication features atop of standard wireless technologies (including Wi-Fi hotspots, Bluetooth, BLE). Nearby connections provide features like automatically turning on Bluetooth or Wi-Fi and authentication via automated background process which enables a fully automated connection process, from peer discovery, peer advertise to connection established. We select *P2P_CLUSTER* as the peer-to-peer strategy that supports an M-to-N, or cluster-shaped, connection topology, which is ideal for building a mesh network for swarm cycling.

In our swarm network model, we describe individual cyclists as nodes. Each cyclist is equipped with a smartphone on which *Bikerider* is installed. A swarm is considered as a connected mesh network, comprised of nodes. When a node is in free state, which means the node does not belong to any swarm and is ready to join a swarm, the node keeps advertising itself. If another node is within the radio range, they are able to connect to each other and synchronize. A swarm forms if there are two nodes or more. A node in the swarm is set to connected state. When a node is away from the swarm and out of the radio range, the node will disconnect from the swarm and be set to free state again.

The nodes are synchronized when they are connected. When the network changes such as nodes join in or leave, the update will also be synchronized within the network through message broadcasting. The network's topology changes in a decentralized way as the number of nodes increases. We set the maximum numbers of simultaneous connection per node to 3 in order to maintain a stable connectivity of the network by avoiding overload connection to a single node. The connections of the network are formed in a 'string-like' way. A single node in free state is considered as a string of length 1. The first node and the last node in the "string" will scan and advertise at the same time and be able to connect to a new "string". In the end there will be a long chain of connection that connect all nodes. Information about the list of connected nodes in the "string" will be synchronized within the "string" in order to prevent a self-loop connection.

In Wireless ad hoc network, connectivity and coverage area the two biggest issues [10]. In swarm cycling, bicyclists are cycling with each other with a safe distance around 3 meters, therefore the coverage of wireless signal is sufficient in this case. However, signal distortion or noise can still cause unexpected temporal disconnection. To prevent a false swarm scenario detection, we used a soft decision method in the *Bikerider*, which detects swarm scenario based on the network connection activities within a time interval.

C. Smart wearables

In order to support the interaction within the swarm or between the swarms in an intuitive way, we developed a wearable LED matrix that can be attached on the back of cyclists. The wearable is intended to visualize various group scenarios like join swarm, without distracting other road users from the traffic situation.

As seen in Figure 2, an ESP32 microcontroller on a DevKitC V4 and the integrated Bluetooth Low Energy interface are used to communicate with the Android application. In addition, the prototype has an ADXL345 accelerometer connected to the microcontroller via I2C protocol in order to detect medium to heavy braking on a level road surface. The LED matrix is built of individually addressable RGB color pixels (namely the WS2812b) and contains 18 x 14 pixels on 29 x 21 cm. The whole system is powered by two power banks with two USB ports per power bank.

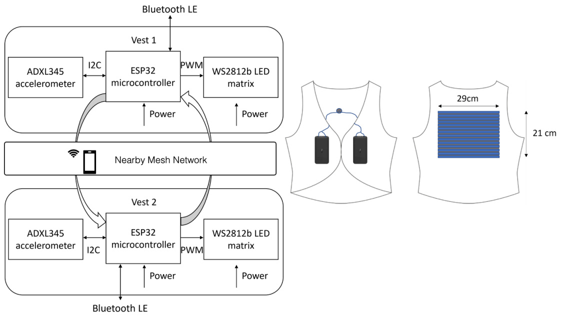


Fig. 2. Prototype setup with system structure

The ESP32 is programmed using the Arduino IDE. It establishes a generic Bluetooth low energy connection to a smartphone, processes the accelerometer readings in order to implement a smart breaking light and generates the patterns to be displayed on the LED matrix using the FastLED library[5]. The smartphone generates a notification based on the output event and sends the notification coded as a number to the microcontroller using the BLE connection. The implementation on the microcontroller generates an interrupt on arrival

[4] https://developers.google.com/nearby/connections/overview

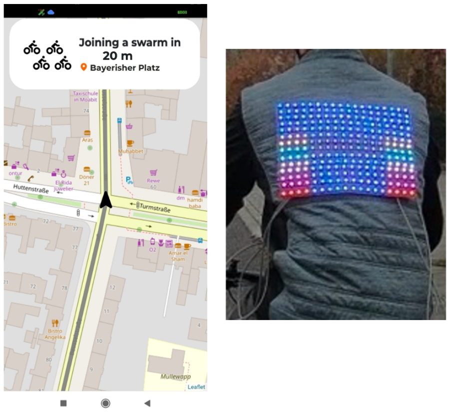
[5] http://fastled.io/

of this number which is used to generate the signal pattern belonging to the notification. The different processes inside the software are grouped into tasks to make use of the two cores of the ESP32 and thus be able to get accelerometer readings and processing them while displaying some notification on the LED matrix or communicating with the smartphone.

IV. SCENARIOS IN SWARM CYCLING

In our work, we use the situational contexts of the user such as the location (GPS coordinates) and the peer-to-peer connection state as input into the system. Regarding the location context we utilize geofencing[6] to generate corresponding system actions and outputs for different use cases. Geofencing allows the system to trigger activities when user enter or exit a radius of given a location which is specified with latitude and longitude. In *Bikerider* we use a geofencing service of Android[7] and set the radius of geofencing to 100 meters considering the location accuracy in outdoor areas according to the recommendation of Android. With geofencing and the peer-to-peer connection state of the nearby mesh network the system will trigger different events according to different scenarios:

- ***Approach swarm*** As mentioned in the Section 2, each beginning waypoint of the trip intersections is denoted as a check point, where the cyclist is supposed to join the cycling swarm. Once the cyclist enters the waypoints that is 100 meters before each check point along the approach swarm event is triggered. As the output of the event, a notification with text information of relative distance between the user and the potential swarm group and the remaining cycling time to the check point will be shown on the smartphone and a signaling notification with a pattern of single color in a pulsing mode with low bright level will be shown on the smart wearable.
- ***Join a swarm*** If a cyclist is in proximity with a swarm group or another bicyclist, or more specific, if the cyclist enters the nearby mesh network radio range, the network connection will be established. The updated swarm information is then synchronized within the swarm group including the new joined bicyclist. This will trigger the system event join swarm. The new joined bicyclist and the other group members will both be notified with a text notification on the smartphone and a signalling notification with two dimmed rows moving from the bottom of the matrix to the top of the matrix, while the other rows maintain their brightness on the smart wearable (see Figure 3).
- ***Leave a swarm*** If a swarm member leaves the proximity of the others, the peer-to-peer connection of the left bicyclist will be lost. This will trigger the event leave swarm besides the cyclist who left the swarm. For the rest of the swarm members, the swarm information are updated and synchronized within the group. In this case, similar as in the join swarm event, they will both be notified with text notification on the smartphone and a signaling notification with two dimmed rows moving from top to the bottom of the matrix on the smart wearable.
- ***Reached critical mass*** If a join swarm increases the group size to more than 15 members the so-called critical mass is reached. Additional colored stripes are displayed on the sides.



Joining a swarm in
20 m
Bayerisher Platz

Fig. 3. Left: Show a notification on the smartphone when bicyclist joins a swarm. Right: Signalling on the LED matrix when a cyclist is joining the swarm (which has more than 15 members)

V. CONCLUSION AND FUTURE WORKS

We introduced connected swarm cycling, a social cycling concept supporting people cycling together with a common direction or destination. A major goal is to increase cycling safety in urban areas by adopting the "safety in numbers" paradigm. We assume that the concept of Swarm Cycling can significantly change the mobility behaviour of citizens and will be a building block of green mobility for sustainable cities in the future. We described the underlying technical concepts and implementation aspects in terms of system architecture, Nearby Mesh Network topology algorithm and the smart wearable prototype. In our actual research project *SocialWear*, we will further extend this technological basis and address the impact of V2X real time communication in terms of dynamic changes within or between swarms, for example, swarms merging or splitting. We will also further explore the user group behaviour models to learn the social group membership of swarm participants and knowledge about group behaviour with user models data. By applying edge-based computing for user model and sharing the models over the Nearby Mesh Network, user behaviour classification and clustering and the generation of group behaviour features can be implemented on device. Furthermore, the learnt models can be used to explore recommendations for best matching (bicycling) groups.

VI. ACKNOWLEDGMENT

This research is funded by the German Federal Ministry of Education and Research (BMBF) project SocialWear (01IW20002).

---

[6] https://en.wikipedia.org/wiki/Geo-fence

[7] https://developer.android.com/training/location/geofencing

REFERENCES

[1] Jonas Larsen. 2017. The Making of a Pro-cycling City: Social Practices and Bicycle Mobilities. Environment and Planning A 49, 4 (2017), 876–892.

[2] Fabian Küster, Senior Policy Officer, European Cyclists' Federation. "Practitioner Briefings: Cycling. Supporting and encouraging cycling in Sustainable Urban Mobility Planning". September 2019.

[3] Peter Lyndon Jacobsen. 2009. "Safety in numbers: more walkers and bicyclists, safer walking and bicycling". (2009).

[4] Alex Berke, Thomas Sanchez Lengeling, Jason Nawyn, and Kent Larson. 2019. Bike Swarm. In Conference Companion Publication of the 2019 on Computer Supported Cooperative Work and Social Computing (Austin, TX, USA) (CSCW '19). Association for Computing Machinery, New York, NY, USA, 1–4.

[5] Stefan Schaffer, Rene Kelpin, and Norbert Reithinger. 2016. "Longitudinal User Experience of a Mobile Service". In PQS 2016 5th ISCA/DEGA Workshop on Perceptual Quality of Systems. Perceptual Quality of Systems. ISCA archive, 132–136.

[6] Wendy Ju and Larry Leifer. 2008. "The Design of Implicit Interactions: Making Interactive Systems Less Obnoxious". Design Issues 24, 3 (2008), 72-84.

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] S. Céspedes, J. Salamanca, A. Yáñez and D. Vinasco, ”Group Cycling Meets Technology: A Cooperative Cycling Cyber-Physical System,” in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 8, pp. 3178-3188, Aug. 2019, doi: 10.1109/TITS.2018.2874394.

[9] O. M. Abu-Sharkh and Z. Dabain, "GreenBikeNet: An intelligent mobile application with green wireless networking for cycling in smart cities," Mobile Netw. Appl., vol. 21, no. 2, pp. 352–366, Apr. 2016.

[10] S. Jadhav and K. N. Nagesh, ”Impairment Impact on the Wireless Communication System,” 2021 2nd Global Conference for Advancement in Technology (GCAT), 2021, pp. 1-5, doi: 10.1109/GCAT52182.2021.9587620.