# Self-Aware Autonomous City: From Sensing to Planning

Bo-Wei Chen

***Abstract*—This article presents a knowledge mining model, where a city can plan its development based on existing knowledge during city expansion, for example, telecommunication resource allocation and crowd forecast in a new region. Unlike most works that focused on Internet-of-Things (IoT) sensing, this study is aimed at urban planning by using harvested data, from the perspective of city architects. For large-scale metropolitan areas, a massive amount of data is generated every day, either from static surveys or dynamic IoT sensing. For urban planners, data collection is not their prior concerns. How to transfer harvested knowledge from exiting parts of the city to suburban/rural/ untapped areas is a new challenge. This is because those areas still lack sufficient statistics, and the density of IoT deployment is low. Therefore, development is risky and uncertain. To exploit new regions requires knowledge inference. Such a transition needs data interpretation from historical city dynamics, involving sensor deployment, human activities, and resource allocation in the vicinity. With the proposed model of this article, a city can estimate the requirement for resources when the peripheral areas on the outskirts of a city develops. The same model can be applied to enterprise sides for resource deployment, and applications are not merely limited to governments.**

***Index Terms* — Knowledge mining, crowd forecast, crowd modeling, data inference, city dynamics, city economics, city decay, city resilience, city monitoring, nonnegative matrix factorization, self-organized sensing, self-organized map, resource allocation, resource planning, mobile edge computing, smart city**

## I. Introduction

City economics, or urban economics, discusses the interplay between various crowd behavior and living environments, e.g., populations, house prices, and communication coverage, in a urban spatial structure. Usually, the methodology for city economics involves two parts. One is detection of urban dynamics, and the other is prediction of urban growth/decay. For detection, typical approaches rely on static surveys, such as questionnaires and household statistics. Today, ever since cyber-physical systems arise, modern city economics no longer passively depends on static data but proactively collects them by leveraging the various Internet of Things (IoT). With advances in miniaturization of electronic circuits, the IoT can be deployed almost in every corner of a city. For example, generic IoT devices like particulate matter and carbon dioxide meters can be installed outdoors to measure the air quality of a city. Gaseous and pH sensors can be respectively placed underground or underwater to monitor sewage. More recently, as artificial intelligence technology has made considerable progress, it subsequently empowers “edge artificial intelligence” — Artificial intelligence for edge computing. Edge computing is an architectural shift from clouds to terminals. Such a shift involves comprehensive technical changes in computation, communications, middleware, and hardware. Edge devices can provide basic functionalities, such as identification and encoding, to alleviate computational loads of cloud sides and constrained bandwidth. Edge artificial intelligence is particularly aimed at pattern recognition or data analysis. For example, edge devices like visual IoT nodes are capable of identifying whether or not sources contain target data. If confirmed, the sources of interest are transmitted to clouds or microdata centers. When harvested data require compression or privacy protection, encoders are enabled before transmission to cloud sides. The functionalities of various edge devices can be dynamically enabled, depending on applications. When a group of devices gather together, they can coordinate each other and reconfigure functionalities to complete a complex task. When edge computing gradually evolves into mobile edge computing, the IoT no longer stays fixed in one place as before. For mobile edge devices, heterogeneous sensors are simultaneously installed in the carrier, like autonomous multirotor drones (i.e., unmanned aerial vehicles) and self-driving cars (i.e., unmanned ground vehicles), to capture various data. These carriers can provide high mobility for city monitoring by forming a fluid, movable, and wireless sensor network. The number of participatory carriers can also be adaptively readjusted anytime. For instance, if city regions have high population densities, new carriers join the network to provide more samples. Sensing therefore becomes more dynamic, and the immersion of IoT sensing can deeply fuse into daily life. Among all types of IoT sensing, drones particularly attract significant attention in recent research. Compared with ground vehicles, drones are capable of adapting to various terrains that are inaccessible to ground vehicles due to aviation capability. Sensing coverage is therefore largely enhanced. In addition to sensing, telecommunication can also be attached to this fluid vehicular network and establishes an ad hoc net, such as Vehicle Ad Hoc Networks (VANETs) and Mobile Ad Hoc Networks (MANETs). For drones, VANETs then become Flying Ad Hoc Networks (FANETs). At present, a great deal of effort has been devoted to urban sensing, for instance, [1-4]. Apart from IoT

B.-W. Chen is with the Department of Electrical Engineering, National Sun Yat-Sen University, Taiwan

approaches, urban monitoring based on crowdsensing [5] and satellite remote sensing is also another widely used method for detection of urban dynamics [6, 7]. As a whole, city dynamics can be captured from different perspectives with the current state-of-the-art IoT. The point is how to utilize and convert these harvested data into knowledge, especially when a city expands. How many base stations are required for new regions when telecommunication service providers operate? What is the influence of the crowd activity? Does the crowd activity cause migration of current arrangement? Is there a transport hub? A key to the above problems resorts to crowd prediction.

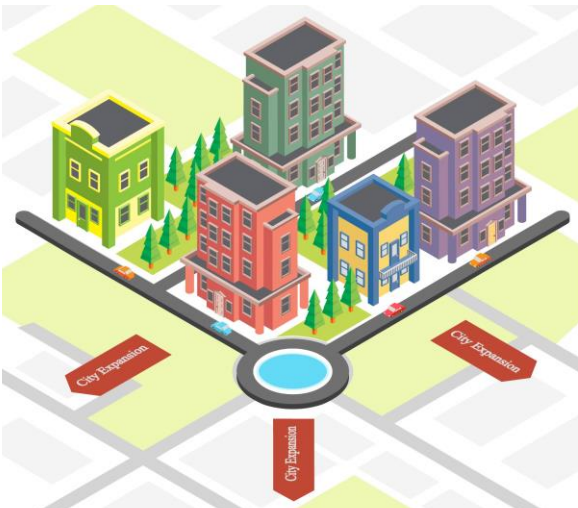


Fig. 1. This article shows how to perform crowd prediction and urban planning in untapped regions when a city expands. Such a prediction can provide clues for resource allocation in advance (parts of the images were designed by Freepik).

During city development, different types of crowds may gradually form, and crowds may vary from street to street and from block to block, depending on crowd types. Herein, crowds refer to all the activities generated by humans, e.g., data traffic and subscribers. Each type of crowds has its own characteristic. Not every region has the same statistic. When more activities occur, crowds and flows become large, subsequently forming hotspots. Crowd activities are usually volatile and fluctuated over time. Beside, different types of crowds are highly diversified.

In practice, it is impossible for a city to grow infinitely when new immigrants keep moving in. The resources could be depleted, and living quality may degenerate. When parts of a city decay or oversaturate, expansion or crowd relocation occurs on the outskirts of a city. Flows may be diverted from existing regions to suburbs. For these peripheral suburban/rural areas, the sampling data of the previously mentioned IoT methods for crowd detection are quite limited. Thus, knowledge inference for those areas may be inaccurate, and development is filled with uncertainty. For city planners and enterprises, uncertainty means risks. How to effectively predict the demand of those new regions is of prior concerns.

Take telecommunication networks for example. The statistics of data flows are a good indicator during deployment of base stations. They are built to balance communication loads and to provide access points to end users (EUs). In a metropolitan area, site selection should consider both service coverage and maintenance costs while constrained communication resources are satisfied at the same. An equilibrium is reached after a long period of adjustment. When a city expands, suburban/rural areas gradually evolve. The scale of the city increases, but original resources no longer support the scale. With those developing areas, the equilibrium and activity hotspots may shift. To evaluate the influence of newly emerged crowds and discover activities in advance is the topic addressed in this article.

## II. Fluid Mobile Sensing Based on Self-Organized Swarm Intelligence

To be self-aware, the first step is to collect as many data as possible before the self-reasoning process is performed. The following sections introduce a swarm intelligence-enabled sensing technique for mobile edge devices.

.

### *A. Self-Organized Maps*

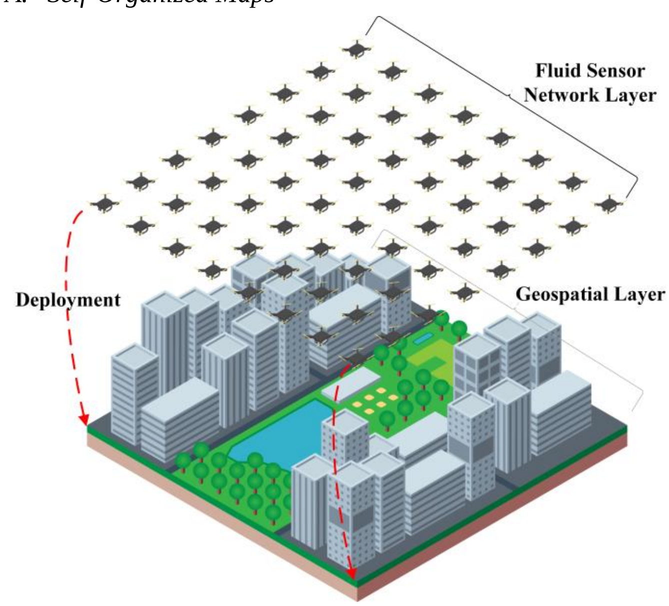


Fig. 2. Self-organized mobile edge array, where the topology can be dynamically changed based on crowds. The mobile edge array is composed of mobile edge devices, each of which is connected to proximal devices. The sensor array is deployed in the city (parts of the images were designed by Freepik).

The self-organized map (SOM) is a type of neural networks, proposed by Teuvo Kohonen. A typical SOM consists of only two layers of neurons. One is the input, and the other is the visual map space. Neurons between layers are interconnected by edges. The input layer is used to perceive outside data, whereas the map space reflects the input by changing the topology or appearance. SOMs are usually featured as a visualization tool because of the capability of mapping high-dimensional input data into low-dimensional intelligible images.

Unlike neural networks with multilayered perceptrons that use both feedforward and backpropagation algorithms, SOMs adopt merely the former one. This means changes of the topology in the map space do not loop back to affect the input layer or weights. It is worth noting that each neuron in the map space is fully connected with all of the neurons in the input layer. To reflect strength of connectivity, a value is assigned to each connection. Such a value is the weight between two connected neurons. No weighted edges exist in the same layer.

As a visualization tool where high-dimensional data are transformed into the topology of neural connections, the intuition behind SOMs is simply based on the law of attraction. The map space of SOMs is like an elastic grid net. The node on the net represents a neuron, and the edge denotes the distance between neurons. At first, the neuron on this grid net is uniformly distributed, with a fixed separation between neighbors. Every time when a stimulation inputs, only one neuron on this net is selected. Subsequently, this neuron attracts its neighbors, and the proximal neighbors move closer to the selected neuron. The edges and nodes do not disappear during the entire process. Additionally, no new edges or nodes are appended to the existing SOM, but edges may become shorter or longer. Shorter edges represent higher similarity between two adjacent nodes, whereas longer edges denote lower similarity. In SOMs, attraction assimilates neighbors. This reveals that the selected neuron does not differentiate neighbors, but makes them similar to itself. After self-organized mapping, like attracts like. Groups of similar neurons gather together. This is how SOMs display high-dimensional inputs by using readable mapping.

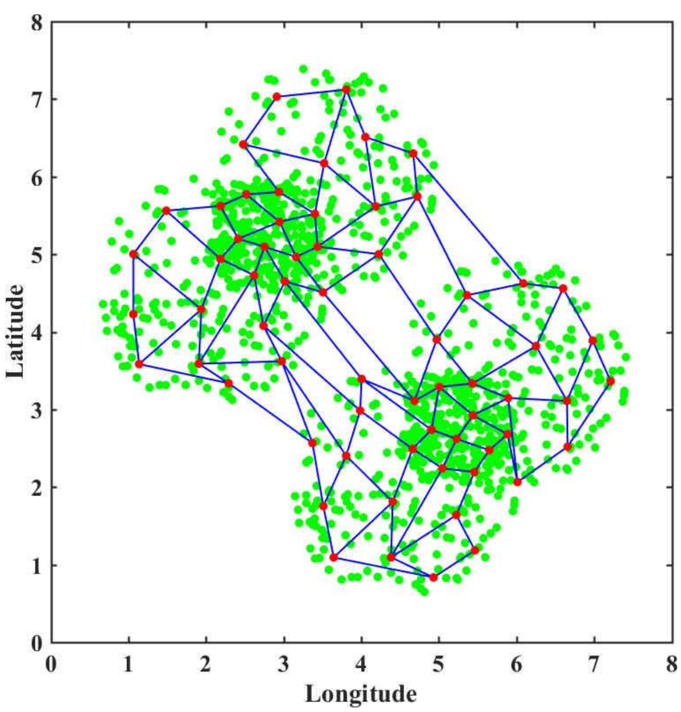


Fig. 3. Self-adaptation of an 8×8 mobile edge array to various end-user (EU) requests. Red dots are mobile edge devices, and blue lines are connections between devices. Green dots represent EU requests.

## B. Self-Organized Mobile Edge Array

In the applications of mobile edge devices, the internal mechanism of SOMs is still the same. The difference is that the neuron in the map space is replaced by an array of devices. The input layer becomes a communication layer that can receive the request from end users (EUs) and commands from dispatch centers (if mobile edge devices for telecommunications are used for example). Whichever device receives a request, it broadcasts to the entire array. The array can dynamically modify its topology to fit the area of interest (i.e., the target area) and the density of requests. With robust control, the array can maintain its shape to provide stable services.

The position of a mobile edge device is mapped into a geographical coordinate — Latitude and longitude. At the initial stage, the topology of an array is a uniform formation. The perimeter of the array approximately covers the area of interest. When more requests are received, the topology of the array gradually updates itself according to the geographical distribution of EUs and the frequency of requests. To make the array adapt to EU requests, the request content and neural weights should contain the geographical coordinate of an EU.

The topology of the array reveals important clues for MANETs. 1) The distribution of mobile devices is based on the population of requests. When the density of the population is higher, more mobile devices focus on the area. This implies that more ad hoc nodes provide resources. 2) The communication load of each device is displayed on the hit map of the SOM. Hit maps collect the statistics of a neuron when it is selected after requests are input. 3) The geospatial distribution of EUs can be estimated by the area of a triangle formed by three devices. As mobile edge devices have GPS coordinates, the area of a triangle is computed based on the cross product of two vectors. 4) The edge of the topology shows the original communication path between connected mobile devices. Devices can identify neighbors by checking these edges. 5) The number of connected edges, or node degrees, indicates whether or not a mobile device is a boundary node. Identification of boundary nodes and nonboundary nodes is important when SOMs needs expansion.

## C. Incremental Self-Organized Mobile Edge Array

The initial size and the shape of the topology usually follows default settings. When the area is larger than expected or when EU requests increase, the existing array may be incapable of offering sufficient resources. This is the reason why SOMs needs expansion, and new mobile devices are required. Furthermore, EUs may relocate, and the perimeter could become boarder than that in the early phase. Although SOMs can update themselves to reflect relocation, a large perimeter lowers efficiency of communications.

Under such a circumstance, growing self-organized maps (GSOMs) [8] are suitable for expansion of the mobile edge array. Unlike typical SOMs, the size of GSOMs can be dynamically expanded. This means new mobile devices can join an existing array and share communication loads. When resources are depleted, dispatch centers can send more mobile devices to a target zone of high population density. As long as the number of nodes in the map space increases, the topology of GSOMs can readjust itself. The target zone can be serviced by new mobile devices and their neighbors. There were several variants of GSOMs. Basically, they can be classified into two types. One supports internally/externally growth [9]. Either boundary nodes or internal nodes allow new nodes. This type has more flexibility. However, reorganization of the topology requires much effort.

The other is external expansion that adds new nodes only at the border, but with lower complexity [10]. Boundary nodes display the location where new mobile devices can join the array. This type does not require rerouting connections of existing nodes. A simple connection with boundary nodes finishes expansion.

### *D. Incremental Hierarchical Self-Organized Mobile Edge Array*

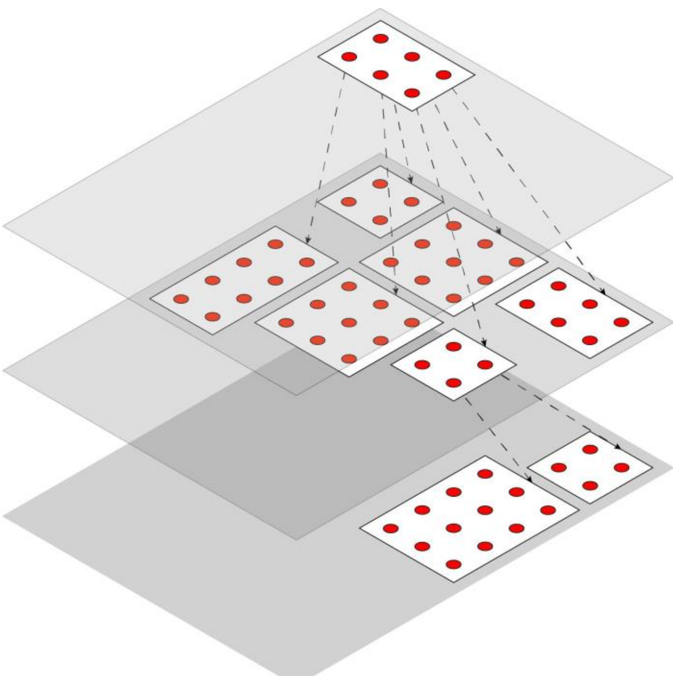

Fig. 4. Hierarchical structure of a three-layered GHSOM. Red circles represent neurons in the map space. Dashed lines are used to indicate the relation between parent and child GSOMs. GHSOMs contain terminals and bridges (or hubs). These two types of neurons can isolate heterogeneous networks when software-defined networks (SDNs) are implemented. For a mobile edge array, the structure of GHSOMs represents a logical formation, not a physical one.

In a typical SOM, there is no difference between one neuron and the other ones. Every neuron is treated equally in the same way. This means communications via a mobile edge array have the same priority, and various collected data are processed with the same scheme. Nonetheless, edge-device sensing could generate data with a variety of rate-distortion characteristics, such as high-definition photographs, acoustic signals, video streams, and environmental readings. This implies multimodal data may be blended with complex transmission requirements, e.g., multisources, multidestinations, multirates, and time sensitivity, which make dispatch centers difficult to manage the network. Moreover, part of the array can be deployed in diversified terrains to collect different types of data, and each cluster of devices is responsible for various tasks, depending on surveillance missions [20]. Therefore, heterogeneous sensing in FANETs requires fine network management and a hieratical structure.

For such a problem, the growing hierarchical self-organizing map (GHSOM) [11, 12] is a feasible solution to heterogeneous sensing networks among MANETs. GHSOMs are variants of GSOMs, where the topology can be vertically and horizontally expanded. An example is shown in Fig. 4, where three layers of GSOMs are presented. The top layer is the initial layer. The lower layer is derived from the higher one. Directed edges are used to indicate the relation between parent and child layers. The head or the terminal vertex with an arrow points at child layers, whereas the tail or the initial vertex represents source nodes from parent layers. Whether or not a node in the parent layer grows a child layer relies on the learning/adaptation phase. This means not every node in the parent layer points at a GSOM in a child layer. Furthermore, it is worth noting that an arrow points at a GSOM instead a node. A layer without any child layer is a leaf layer.

The topology of each layer and each SOM can be dynamically expanded, depending on the learning/adaptation phase. In fact, GHSOMs can be viewed as nested GSOMs. The node of an outer GSOM can accommodate an inner GSOM. When a node in an outer GSOM has an inner GSOM inside, this node becomes a bridge. Communications across GSOMs or layers must go through bridges.

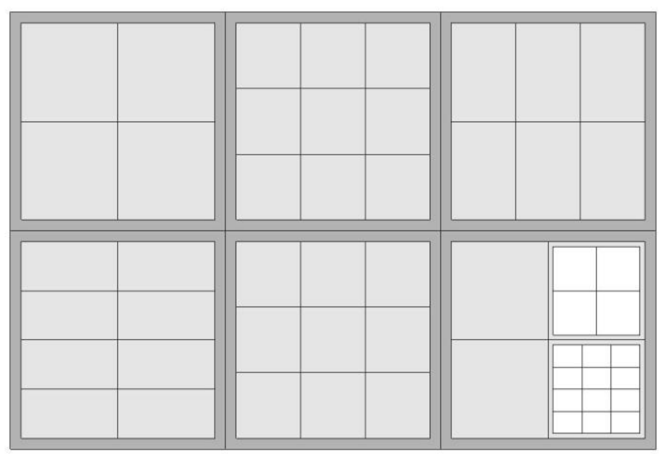

Fig. 5. Nested GSOMs, where each rectangle denotes a logic neuron. A rectangle contains nested rectangles. Different colors are used to represent neurons in various layers. Hierarchical GSOMs are equivalent to nested GSOMs. The structure of this nested GSOM is the same as that in Fig. 4.

With the hieratical structure of GHSOMs, MANETs can be divided into subnets to meet diverse purposes. When all the mobile devices in the same GSOM are of the same type of data sensing or communications, they are homogeneous. The individual GSOM in the leaf layer is typically considered homogeneous. If two separate GSOMs are homogeneous and derived from the same parent GSOM, the parent GSOM becomes heterogeneous. In highly diverse heterogeneous networks, where homogeneous and heterogeneous GSOMs coexist in a MANET, bridges are important. They can isolate heterogeneous subnets and control flows (e.g., prioritizing data streams).

Based on this concept, the communication functionality of mobile edge arrays at least has two modes — Terminals and bridges. Terminals are for data sensing. Bridges coordinate subnets and prioritize transmissions. Besides, bridges can also work as sink nodes for data pooling before collected data are transmitted to cloud centers. Modern algorithms have feasible solutions to hub selection for efficient data delivery [23, 24].

To fulfill GHSOMs in a mobile edge array, software-defined networks (SDNs) are applicable. SDNs use software rather than traditional hardware-based methods to control data flows [25-27]. Various functionalities can be dynamically enabled by virtualization (like virtual machines) [28], based on different tasks or applications. As mentioned earlier, both terminals and hubs are required in a GHSOM. However, whether a device is a terminal or a hub relies on GHSOM inputs. This implies the internal system of devices should be capable of dynamically reconfiguring itself into different modes that undertake either Edge Computing for remote sensing [29, 30] or soft switching for data transmission. SDNs have such flexibility that is suitable for nonstatic heterogeneous networks [31]. In brief, with the use of SDNs, a drone can become a terminal and a hub, depending on Growing SOMs. The process is automatic and not defined by dispatch centers.

## III. City Resiliency Management by Prediction and Planning

As the peripheral suburban/rural area on the outskirt contains incomplete information on crowds, one of incomplete data analyses "collaborative filtering (CF)" is introduced in the following subsections.

### *A. Multiple Indicator Planning*

Collaborative filtering is a technique for missing-value imputation. Imputation means to fill in entries that contains missing values with approximate values. At present, CF has developed many approaches, such as item-based methods and matrix factorization. This article particularly concentrates on the category of matrix factorization, where nonnegative matrix factorization is examined herein, because it fits urban monitoring applications. Nonnegative matrix factorization works on matrices, and it decomposes a matrix into two matrix factors, where elements are nonnegative. With criteria like Frobenius norm, nonnegative matrix factorization is capable of handling matrices with missing values. In urban planning, the layout of a city can be converted to a planar map. Based on this map, the entire city including suburban/rural regions can be divided into rectangular grids. Each grid may contain various statistics from urban sensing. Typically, one representative value is selected for a grid. For crowd prediction, the process is the same as imputation. Given a grid view of a city map $\mathbf{X}$, nonnegative matrix factorization can decompose $\mathbf{X}$ into two factors $\mathbf{W}$ and $\mathbf{H}$, such that $\mathbf{X} \approx \mathbf{WH}$. At the initial stage, zeros are placed in those entries with missing values to enable arithmetic computation. Meanwhile, $\mathbf{W}$ and $\mathbf{H}$ are initialized with random nonnegative numbers. The imputation is based on minimizing Frobenius norm of the error, or equivalently the distance between $\mathbf{X}$ and $\mathbf{WH}$. To minimize errors, the decomposition employs an iterative process, consisting of two phases. One is element-wise computation of Frobenius errors, and the other is the element-wise gradient update of the two factors. Both phases disregard the entries with missing values and focus on complete ones. When the iteration converges, those missing values can be imputed by simply taking the product of $\mathbf{W}$ and $\mathbf{H}$.

With collaborative filtering, city planners can evaluate and predict the statistics of new regions. Such data can help city planners, for example, determine whether or not a new base station should be built in a new region. The following figure shows an example illustration with three types of crowd behavior in communications — Service data (i.e., telecommunication controls), data traffic, and the number of subscribers [13]. Let $L$ represent the total number of crowd types. Then, $L$=3 in this case because three types of crowd activities (i.e., $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{X}_3$) are used for modelling. Furthermore, $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{X}_3$ are respectively for service data, data traffic, and the number of subscribers.

In Fig. 6, each type is denoted as a layer composed of grids. A grid contains estimated statistics with respect to one type of crowd activities. Notably, these three layers are separately processed in this example. Therefore, three sets of factors are generated, with each set corresponding to its type of crowd behavior.

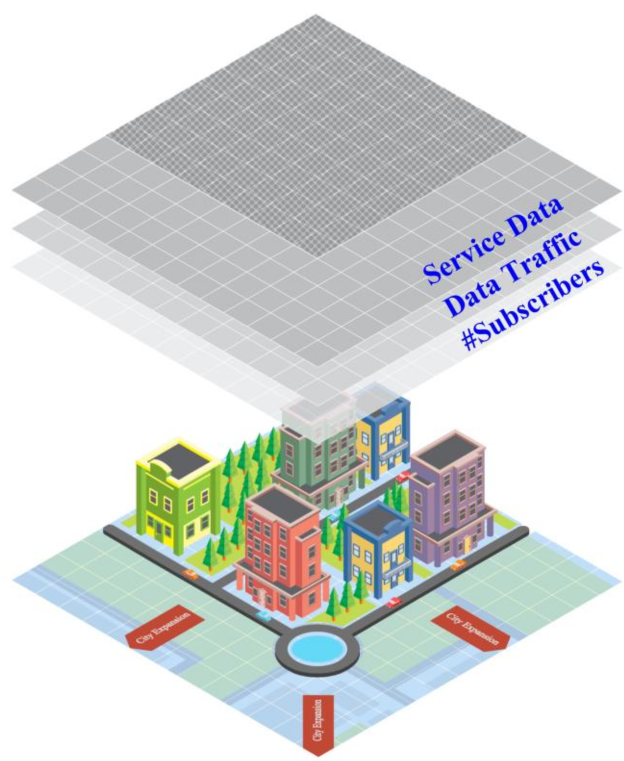


Fig. 6. Crowd prediction based on context-aware CF. Sample illustration with three types of crowd activities are displayed in the top three layers. The dotted grids in the top layer is the existing developed city regions. The other grids

outside the dotted ones are new urban regions (parts of the images were designed by Freepik).

After prediction/analysis by CF, city planners can make arrangements in advance by subsequently examining SOM simulation results. For instance, assume CF predicts that more EU requests appear in the new region, as shown in Fig. 7. SOM simulations on such a new distribution can reveal the movement of existing mobile edge devices after the new EU requests are input. Fig. 7 shows that the lower cluster of the edge array slightly moves towards the new region to maintain an equilibrium. City planners can determine whether or not the new coverage can satisfy the growing EU requests.

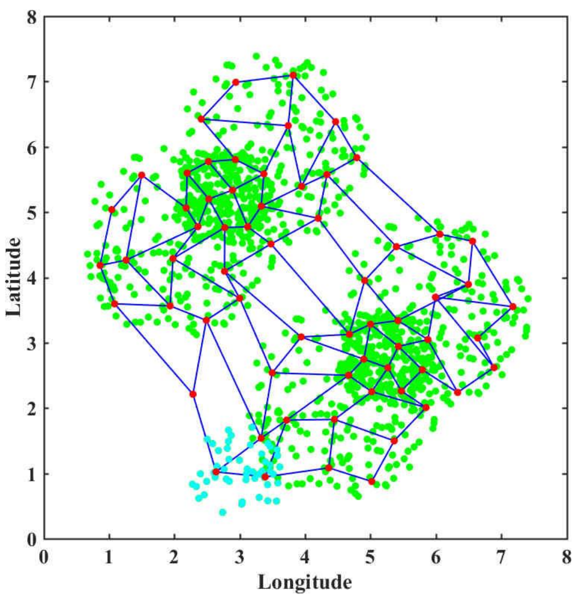

Fig. 7. Incremental self-adaptation to newly added EU requests. Pale blue markers represent new EUs in a new urban region. City planners can examine the SOM result to make arrangements for resource allocation in advance.

### B. Time-Series Planning

To further estimate crowd activities over time, one can introduce time-series statistics of crowd activities for predictive analysis on new regions. Based on the input sequence $\mathbf{X}[t]$, where $t$ signifies the time index (e.g., daily, weekly, monthly, and annually), the estimated statistic for the target region can present a time-series characteristic. When planning considers multiple indicators (i.e., $\mathbf{X}_\ell[t]$ where $\ell$ represents the index of crowd types), more than one time-series curve is present. By cross-examining the rise and fall of these curves, city resiliency can be quantified.

For telecommunication service providers, supplies and demands are always an optimization problem on profits and costs. Revenues come from EU subscribers, and available resources are based on operational costs. Unlike typical research that emphasized optimization based on current observations, CF provides a prospective view on future changes and planning.

### C. Heterogeneous Indicator Co-Planning

In multiple indicator planning, different types of statistics are separately processed in CF. However, indicators could be interconnected and influenced by each other. For example, the number of subscribers could be affected by pricing and populations. Dependent variables are constrained by independent variables, or covariates. Individual CF on dependent variables does not reflect such a relation. For complete data, the relation between dependent and independent variables has been widely studied in statistics, and techniques range from ordinary least squares to factor analysis. Nevertheless, for incomplete information, all the variables in new regions are unknown. How does CF work?

Multimodal CF [14] was designed for such a purpose, where $L$ types of $\mathbf{X}_\ell$ ($\ell$=1,…,$L$) are coprocessed at the same time. Relations between matrix factors are imposed as constraints during decomposition. After multimodal CF, total $L$ sets of matrix factors are generated. With multimodal CF, the predicted statistics of new regions consider collinearity. This implies predicted statistics do not diverge or contain unreasonable values.

### D. Flow Capacity Planning

The above-mentioned planning is primarily based on regions or blocks. For modeling traffic flows or those activities that cross geographical boundaries, e.g., data flows between hubs and terminal devices, connections instead of end points are the main focus. In such a scenario, flow prediction brings more valuable information than regional statistic estimation. Take base stations for example. If data flows from new regions to each stations can be estimated in advance, whether or not new stations should be built can be arranged. Site selection can be estimated among multiple regions. To satisfy such a purpose, graphical models are a useful tool for flow capacity planning. In graphical modeling, activities are represented as a graph, and only edges and vertices are used to form a graph. Edges denote connections and vertices indicate destinations or sources. Activities like vehicular traffic, crowd flows, transportation, and communications are suitable for graphical modeling.

To perform CF on graphs, typical adjacency matrices are used. A graph G = <V,E> with vertex set V and edge set E can be converted into an adjacency matrix $\mathbf{A}$ by using a $|V|\times|V|$ array. Operator $|\cdot|$ signifies the size. In adjacency matrices, nonzero elements indicate the presence of edges. In other word, when two vertices $i$ and $j$ are connected, $\mathbf{A}(i,j)$ and $\mathbf{A}(j,i)$ are both nonzero. For the other elements, the values are all zeros if no edge exists. Weighted edges are applicable when capacities of connections vary.

In flow capacity planning, when new vertices join the existing graph, the adjacency matrix expands. Whether or not there are connected edges between new vertices and existing ones depends on CF results. CF on graphs can estimate the amount of flows between vertices. When graphical CF is combined with time-series information or heterogeneous flows, more functionalities are provided for city planners.

## IV. Conclusion

This article presents an urban microeconomic model — How knowledge of city dynamics can be adaptively collected and used for planning. The former relies on swarm intelligence, where a fluid mobile edge array is adopted for self-organized sensing. The latter utilizes collected data and context-aware CF for resource/crowd prediction during city expansion. The prediction can be fed back to the swarm intelligence model for validating the future influence on resource balances after resource allocation is made. Two models resonate and iterate to discover solutions.

In self-organized sensing, an incremental array of mobile edge devices (GSOMs) and a hierarchical array (GHSOMs) are respectively discussed in the article. Both of them empower data sensing in highly diversified and heterogeneous environments. Regarding city planning, various CF models are used for urban reasoning, including multiple indicator planning, time-series planning, heterogeneous indicator co-planning, and flow capacity planning. They can be further combined together to enhance reasoning capability. Such an urban microeconomic model can benefit both city planners and business operators in strategic making.

## Acknowledgment

The author would like to thank Freepik for contributing icons in Figs. 1, 2, and 5.

## References

[1] P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, “Convergence of MANET and WSN in IoT urban scenarios,” *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3558–3567, Oct. 2013.

[2] Y. Zhu, Z. Li, H. Zhu, M. Li, and Q. Zhang, “A compressive sensing approach to urban traffic estimation with probe vehicles,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2289–2302, Nov. 2013.

[3] R. Du, C. Chen, B. Yang, N. Lu, X. Guan, and X. Shen, “Effective urban traffic monitoring by vehicular sensor networks,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 1, pp. 273–286, Jan. 2015.

[4] V. A. Siris, E. Z. Tragos, and N. E. Petroulakis, “Experiences with a metropolitan multiradio wireless mesh network: Design, performance, and application,” *IEEE Communications Magazine*, vol. 50, no. 7, pp. 128–136, Jul. 2012.

[5] G. Cardone, L. Foschini, P. Bellavista, A. Corradi, C. Borcea, M. Talasila, and R. Curtmola, “Fostering participaction in smart cities: a geo-social crowdsensing platform,” *IEEE Communications Magazine*, vol. 51, no. 6, pp. 112–119, Jun. 2013.

[6] D. Tuia and G. Camps-Valls, “Urban image classification with semisupervised multiscale Cluster Kernels,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 1, pp. 65–74, Mar. 2011.

[7] A. Anjomshoaa, F. Duarte, D. Rennings, T. Matarazzo, P. d. Souza, and C. Ratti, “City scanner: Building and scheduling a mobile sensing platform for smart city services,” *IEEE Internet of Things Journal*, 2018.

[8] M. M. Van Hulle, “Self-organizing maps,” in *Handbook of Natural Computing*, G. Rozenberg, T. Bäck, and J. N. Kok, Eds. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 2012, pp. 585–622.

[9] B. Fritzke, “Growing self-organizing networks — Why,” in *Proc. European Symposium on Articial Neural Networks*, Brussels, Belgium, 1996, Apr. 24–26, pp. 61–72.

[10] D. Alahakoon, Saman K. Halgamuge, and B. Srinivasan, “Dynamic self-organizing maps with controlled growth for knowledge discovery,” *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 601–614, May 2000.

[11] A. Rauber, D. Merkl, and M. Dittenbach, “The growing hierarchical self-organizing map: Exploratory analysis of high-dimensional data,” *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1331–1341, Nov. 2002.

[12] D. Merkl, S. H. He, M. Dittenbach, and A. Rauber, “Adaptive hierarchical incremental grid growing: An architecture for high-dimensional data visualization,” in *Proc. 4th International Workshop on Self-Organizing Maps*, Kitakyushu, Japan, 2003, Sep. 11–14, pp. 293–298.

[13] S. Burnton, “Report of the interception of communications commissioner: Annual report for 2016,” Interception of Communications Commissioner, Technical, Dec., 2017.

[14] N. Seichepine, S. Essid, C. Févotte, and O. Cappé, “Soft nonnegative matrix co-factorization,” *IEEE Transactions on Signal Processing*, vol. 62, no. 22, pp. 5940–5949, Nov. 2014.