Research Article

Utility-Oriented Placement of Actuator Nodes with a Collaborative Serving Scheme for Facilitated Business and Working Environments

Chi-Un Lei,[1] Woon Kian Chong,[2] and Ka Lok Man[3,4]

[1] *Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong Island, Hong Kong*
[2] *International Business School Suzhou, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China*
[3] *Xi'an Jiaotong-Liverpool University, Suzhou 215123, China*
[4] *Yonsei University, Seoul 120-749, Republic of Korea*

Correspondence should be addressed to Chi-Un Lei; culei@eee.hku.hk

Places to be served by cyber-physical systems (CPS) are usually distributed unevenly over the area. Thus, different areas usually have different importance and values of serving. In other words, serving power can be excessive or insufficient in practice. Therefore, actuator nodes (ANs) in CPS should be focused on serving around points of interest (POIs) with considerations of "service utility." In this paper, a utility-oriented AN placement framework with a collaborative serving scheme is proposed. Through spreading serving duties among correctly located ANs, deployment cost can be reduced, utility of ANs can be fully utilized, and the system longevity can be improved. The problem has been converted into a binary integer linear programming optimization problem. Service fading, 3D placements, multiscenario placements, and fault-tolerant placements have been modeled in the framework. An imitated example of a CPS deployment in a smart laboratory has been used for evaluations.

1. Introduction

Cyber-physical systems (CPS) are systems that contain numerous distributed, linked, and autonomously operated sensor nodes (SNs) and actuator nodes (ANs) [1]. Generally, CPS are used to gather adequate information about the physical environment, manipulate cyber/physical/information quantities, and, eventually, provide useful and prompt services for people [2–6]. In order to provide services, ANs interact with the physical environment by means of transducers and actuators.

In order to ensure coverage and credibility of the service, as well as minimizing resources consumed by nodes in CPS, nodes should be placed in correct positions through planning. Comparing to SNs, ANs are usually equipped with better capabilities of control, computation, and action as well as larger battery capacity. In addition, requirements and physical interaction of sensing and serving are different.

Therefore, existing placement algorithms for SNs [7] are inapplicable to be directly adopted for ANs.

Placement of actor nodes (e.g., mobile robots and unmanned aerial vehicles) and mobile ANs has been explored in the literature [8–10]. Since these nodes can travel to the site of the event without obstructions, existing algorithms do not consider the uncertainty in the ability of actuators to deliver adequate service due to disturbance or fading in harsh environments. However, ordinary, small-size, and low-cost ANs are usually located in fixed positions and have different behaviors of motions and physical interactions. Therefore, algorithms for placement of actor nodes are also inapplicable for placements of ANs. Thus, specific placement frameworks for placement of ANs are needed.

Meanwhile, service coverage, system connectivity, and system longevity are typical metrics for node placement problems [7]. However, we believe that ANs in CPS should be focused on achieving a reliable and accurate serving around

points of interest (POIs) with considerations of a "service utility" metric, because of the following.

(i) Places/objects to be concerned/served (e.g., targets in the battlefield) or crucial occurrences of an event are usually distributed unevenly over the area. Thus different areas usually have different importance and values of serving [11].

(ii) Same type of ANs are often used in an application, but serving environments and requirements of each POI can be versatile and different [12]. As a result, serving power can be excessive or insufficient in practice. Thus, cost-effective approaches should be designed to spread serving duties among ANs.

(ii) Balanced serving workload of ANs can prevent overloading of particular ANs beyond their limit output performance. This can avoid aging/deteriorating of equipment on ANs and, thus, improve the system longevity.

These concerns are significant yet have not been explored thoughtfully by researchers. However, in the algorithm, partial serving of POIs and utilization of redundant residual utility in ANs have not been taken into account. It is undesirable due to setup and operation cost of unnecessary ANs. Thus, it is worthwhile to develop a CPS design framework that can fully utilize the capability of ANs via a collaborative serving scheme. However, this issue does not seem to have been explored in great detail by researchers; yet, it is significant not only to many indoor (business) environments, including offices, hospitals, laboratories, schools, business facilities, and factory workshops.

Based on the preliminary study [13], a relaxed utility-oriented AN placement framework is proposed in this paper. In the framework, utility of ANs is fully utilized and the fidelity of providing service to POIs is improved. To summarize, contributions of the framework are as follows.

(1) The utility-oriented placement problem has been converted into a binary integer linear programming problem. In addition, a recursive framework through the refinement of the search space has been proposed for finding a more suitable configuration.

(2) Serving requirements of POIs, serving capabilities of ANs, features of delivered service, and status of serving activities have been modeled in a unified framework through a generic description. In particular, geographical attributes and obstacles can be modeled as service fading, for a more realistic modeling for deployment.

(3) The proposed framework can be applied to placements for fault-tolerant serving, such that POIs are still partially served even if some ANs are damaged.

(4) The proposed framework can be used for multiscenario placements, such that the system can change its serving characteristics according to the dynamic nature of the environment, without relocating ANs or consuming unnecessary energy for actions.

(5) The proposed framework can be used for placements in three-dimensional (3D) fields, such that ANs can be placed on tall objects (e.g., buildings or trees) or for airborne/underwater serving.

(6) An imitated example of a CPS deployment in a smart laboratory has been used for evaluations of the proposed framework.

The paper is organized as follows. Search space and derived objective function as well as constraints for the single-step placement are presented in Section 2. Generalizations for recursive placement are shown in Section 3, respectively. Finally, the performance of the framework is evaluated by examples in Section 4.

2. Single-Step Placement of Actuator Nodes

The objective of the framework is to pick a set of ANs and assign them to serve a set of POIs. In other words, the framework has to determine the location and service coverage radius of deployed ANs and distribution of serving workload among ANs, with a minimized total cost for setup and operation. Here, we assume that ANs can serve different POIs with different amount of utility at the same time.

The placement problem is formulated as a binary integer linear programming (BILP) problem with a cost objective function and constraints that models different concerns in the deployment of ANs. For ease of explanations, the single-step optimization framework is first described in this section. Then, the recursive framework with a refined search space is described in Section 3.

2.1. Search Space in the Optimization. In the framework, I AN candidates can be selected to serve J POIs. For ease of explanations, A_i denotes the ith candidate of AN and P_j denotes the jth POI, where $i = 1, 2, \ldots, I$ and $j = 1, 2, \ldots, J$. The Cartesian coordinates of A_i and P_j in 2D fields are $\{x_{A_i}, y_{A_i}\}$ and $\{x_{P_j}, y_{P_j}\}$, respectively. In 3D fields, the Cartesian coordinates of A_i and P_j are $\{x_{A_i}, y_{A_i}, z_{A_i}\}$ and $\{x_{P_i}, y_{P_i}, z_{P_i}\}$, respectively.

The placement problem without mesh discretization is an integer nonlinear programming problem. To simplify computations, the infinite search space of AN locations is replaced by a search space with preselected finite elements. All AN candidates are located at the grid points, in which the grid size depends upon the accuracy of placement desired. Since the search space grows exponentially with the increase in the number of points, irrelevant candidates are neglected in the optimization problem. This is further explained in Section 2.3.

Through the discretization, the problem becomes a BILP problem. Thus, branch and bound (BB) algorithm can be used to decompose the BILP problem into linear programming (LP) problems. Optimality of the BILP solution in a finite search space is guaranteed by combining the BB framework and the simplex algorithm in LP problems. However, BILP problems are NP-complete [14, 15]. Therefore, only suboptimal solutions can be efficiently obtained via heuristic methods, such as tabu search and simulated annealing. However, it is out of the scope of this paper.

2.2. Cost Objective Function. The objective of the framework is to design a CPS that can serve all POIs and minimize total cost for setup and operations. The cost can be described by the following expression:

$$\min \sum_{i=1}^{I} E \times Y_i + r_i, \tag{1}$$

where Y_i is the binary selection parameter of A_i; that is, $Y_i = 1$ indicates candidate A_i is selected in the placement. Furthermore, E is the setup cost of an AN and r_i is the service coverage radius of A_i. Generally $E := 100$, such that all POIs are served with the minimum number of ANs. Meanwhile, $\{r_i\}$ should be minimized because ANs with a smaller coverage radius can use less power to serve, which leads to a higher energy efficiency and longevity of ANs and the system. Coverage radius can be adjusted through control units and actuators on ANs.

In some situations, for example, early design stages, we only need to find out the minimum number of required ANs for serving ($I_{\min}$), while locations and serving coverage of ANs are unnecessary. In this case, expression (1) can be replaced by a simplified objective function for a quick cost analysis:

$$\min \sum_{i=1}^{I} Y_i. \tag{2}$$

In the second step, specifications of ANs can then be optimized for the best system performance and largest system longevity by the following objective function:

$$\min \sum_{i=1}^{I} r_i \quad \text{with} \sum_{i=1}^{I} Y_i = I_{\min}. \tag{3}$$

2.3. Constraints on the Coverage of Services. In the framework, inspired by Elfes model [16], a relaxed disc service coverage zone centered at A_i is adopted in the framework. In the framework, P_j can be served by A_i if it is within the coverage of A_i, while the serving efficiency depends on the distance between P_j and A_i (further discussions about service fading are given in Section 2.4). This restriction can be described by the following constraint:

$$r_i \geq d_{i,j} \times X_{i,j}, \tag{4}$$

where $X_{i,j}$ is the binary connection parameter between A_i and P_j; that is, $X_{i,j} = 1$ indicates P_j is served by A_i. $d_{i,j}$ is the Euclidean distance between A_i and P_j. In 2D fields, $d_{i,j} = \sqrt{(x_{A_i} - x_{P_j})^2 + (y_{A_i} - y_{P_j})^2}$, while in 3D fields, $d_{i,j} = \sqrt{(x_{A_i} - x_{P_j})^2 + (y_{A_i} - y_{P_j})^2 + (z_{A_i} - z_{P_j})^2}$.

In practice, ANs have a limited service coverage radius, which are based on the capability and power of circuits and devices on ANs. For simplicity, maximum service coverage radius of all ANs is $r_{\max}$. Furthermore, if A_i is not selected, $r_i := 0$. These requirements can be modeled as

$$0 \leq r_i \leq r_{\max} \times Y_i. \tag{5}$$

Meanwhile, if P_j is within the range of A_i, then P_j must be assigned to A_i. This can be modeled by the following constraint:

$$r_i - d_{i,j} \leq r_{\max} \times X_{i,j}. \tag{6}$$

In order to reduce the optimization problem, unnecessary AN candidates are neglected. For example, it is infeasible for an AN to serve a POI which is farther than its serving distance. Therefore, if $d_{i,j} > r_{\max}$ for any i and j; $X_{i,j} := 0$ and $q_{i,j} := 0$. Meanwhile, it is also not practical for an AN located at the same position of POIs or obstacles. Therefore, if A_i is located at the same position of POIs (i.e., $d_{i,j} = 0$) or obstacles, $Y_i := 0$.

2.4. Constraints on Consumption and Generation of Utilities with Collaborative Serving and Service Fading. Insufficient serving power to POIs may fail to provide guaranteed services and eventually deteriorate the performance of the system. Thus, it is necessary to consider the rate/amount of utilities that can be provided by ANs (utility budget) and rate/amount of utilities that are consumed by POIs. To be specific, each POI should receive a collection of required utility u_j by fusing service provided by multiple ANs. In other words, A_i contributes part of its utility budget to serve P_j, and P_j is served by contributions from several ANs. Comparing to [13], by collaborative serving, less ANs are needed to serve all POIs. An example is shown in Figure 1. Assume that after serving a POI, the residual utility of an AN is not able to serve another POI. However, if collaborative serving is allowed, the middle POI can collect enough utilities that are residual utility of its two surrounding ANs. Therefore, one AN is saved for serving. Collaborative serving can be described by the following expression:

$$C \times \left(\sum_{i=1}^{I} q_{i,j} \right) \geq u_j, \tag{7}$$

where u_j is the total consumed utility of P_j, $q_{i,j}$ is the proportion of utility budget of A_i that contributes to P_j, and C is the maximum utility budget of an AN that can serve to surrounding POIs. Meanwhile, serving capability of every AN is limited by its utility budget. Therefore, the following inequality is required in the problem:

$$\sum_{j=1}^{J} q_{i,j} \leq 1. \tag{8}$$

If part of the utility budget in A_i is used to serve P_j, A_i is selected for serving and it is responsible to serve P_j. These can be described by the following inequalities:

$$\sum_{j=1}^{J} q_{i,j} \geq Y_i,$$

$$X_{i,j} \geq q_{i,j}. \tag{9}$$

In practice, the strength of service power usually decays when the service is delivered from the AN to the POI. In other words, an AN can serve several POIs that are close to the
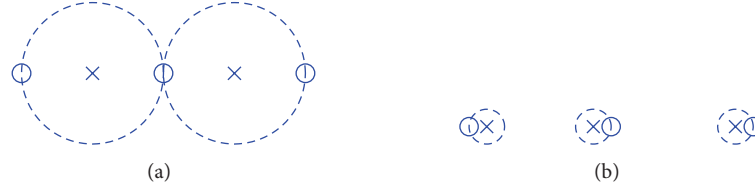
Figure 1: Serving of POIs: (a) if collaborative serving is allowed and (b) if collaborative serving is not allowed.

AN, but it can only serve one POI that is far from the AN. In order to describe this phenomenon, (7) can be replaced by the following constraint:

$$C \times \left(\sum_{i=1}^{I} q_{i,j} \times \left(\frac{f_{i,j}}{d_{i,j} + g} \right)^2 \right) \geq u_j, \qquad (10)$$

where $f_{i,j}$ is the factor for path loss of the delivered service from A_i to P_j and g is the factor for the operation loss of an AN. $\{f_{i,j}\}$ and g and $\{u_j\}$ and C can be determined through product specifications, theoretical estimations, or experimental measurements. In brief,

(i) $\{u_j\}$ depends on scenarios of serving. In particular, it depends on accuracy, reliability, relevance, timelessness, and usability of service. It also depends on whether the service is periodic or event-driven. In general, $\{u_j\}$ should be higher for critical POIs, such that allocated AN(s) will pay more attention to these POIs. For example, the POI may require a larger $\{u_j\}$ if there is a higher possibility of making a significant damage, due to the severity of crashes and shocks.

(ii) $\{f_{i,j}\}$ depends on features and decay rate of the service and the ambient environment as well as the disturbance between ANs and POIs (e.g., trees and walls). In general, ANs can serve targets that lie in their line of sight; therefore, $\{f_{i,j}\}$ should be larger; on the other hand, obstacles may make the service unreachable; therefore, $\{f_{i,j}\}$ should be smaller.

(iii) C and g depend on the battery capacity, power dissipation and characteristics of actuators, onboard circuitry, and peripherals on ANs.

The proposed generic constraint can be further generalized by including specific considerations [17, 18], such as battery models with energy harvesting capabilities [17]. However, the discussion of utility generalizations is beyond the scope of the paper.

2.5. Constraints for Fault-Tolerant Serving and Serving with Limited Recipients per AN.
Generally, every POI is served by the minimum number of ANs. However, due to the impact from environment hazards and physical shock, ANs may be damaged or even failed to operate after deployment. In these dynamic situations, the system is expected to tolerate failures of some ANs and communication link as well as guaranteeing a proportion of serving ability to POIs. Introducing adequate redundant ANs can surely ensure the connectivity. However, it also requires more installed ANs, which can be undesirable

due to the cost. In our framework, fault-tolerant serving can be done by ensuring each POI is served by at least K AN(s), such that at least POIs are partially served by ANs if minorities of ANs are broken. This design restriction can be described by the following connection constraint, for any P_j:

$$\sum_{i=0}^{I} X_{i,j} \geq K. \qquad (11)$$

In the framework, we assume that ANs can serve unlimited POIs at the same time. However, sometimes this assumption becomes impractical because of the hardware capabilities of ANs. In these situations, we need to ensure that each AN can only serve at most L POIs. This design restriction can be described by the following connection constraint, for any A_i:

$$\sum_{j=0}^{J} X_{i,j} \leq L. \qquad (12)$$

2.6. Multiscenario Placement through a Relaxation of Constraints.
In the proposed framework, ANs are placed in fixed locations and serve POIs according to a single scenario. However, serving patterns and requirements can be changed according to the pattern of events as well as application-level interests. For example, requirements of air ventilation during daytime and night-time can be different. In such circumstances, a fixed arrangement is not capable to consider dynamic changes for a better delivery of services. Therefore, in order to improve the feasibility of meeting timeliness requirements, ANs' service should be rescheduled without node relocation. Through the proposed generalization, the framework can place ANs with considering multiple serving scenarios.

Assume that there are K serving scenarios, in order to distribute serving duties to ANs for each scenario correctly, $\{X_{i,j}\}$ and $\{q_{i,j}\}$ in (7)–(11) are replaced by $\{X_{i,j,k}\}$ and $\{q_{i,j,k}\}$, where $X_{i,j,k}$ is the binary connection parameter $X_{i,j}$ and $q_{i,j,k}$ is the proportion parameter $q_{i,j}$, for the kth scenario, respectively, for $k = 1, \ldots, K$. Meanwhile, in order to determine the coverage radius of ANs, $\{X_{i,j}\}$ are used as usual for (4)–(6). Furthermore, A_i is selected even if A_i is only selected in one of the scenarios; that is, $X_{i,j} = 1$ if $X_{i,j,k} = 1$ for any k, and $X_{i,j} = 0$ if $X_{i,j,k} = 0$ for all k. Therefore, determination of $\{X_{i,j}\}$ is enforced by the following linear inequality constraints, such that the generalized problem can still be solved by a BILP optimization process:

$$X_{i,j} \geq X_{i,j,k}, \quad \text{for } k = 1, \ldots, K, \qquad (13)$$

$$X_{i,j} \leq \sum_{k=1}^{K} X_{i,j,k}. \qquad (14)$$

```
(1)  Construct initial search space {A_i};
(2)  repeat
(3)      Determine {d_{i,j}} via locations of {A_i} and {P_j}
(4)      if I_min is not given then
(5)          Determine ∑ Y_i and {q_{i,j,k}} through minimizing (1), subject to (4)–(14) for all i, j and K, via BILP;
(6)      else
(7)          Determine {r_i} and {q_{i,j,k}} through minimizing (3), subject to (4)–(14) for all i, j and K, via BILP, with given I_min;
(8)      end if
(9)      Select {q_{i,j,k}} and {r_i} as the answer;
(10)     Construct the refined search space {A_i} from the determined answer;
(11) until ‖{r_i}_previous − {r_i}_current‖_2 / ‖{r_i}_previous‖_2 ≤ ε
```

ALGORITHM 1: Pseudocodes of the recursive framework.

3. Recursive Placement via a Refined Search Space

The single-step placement framework described in Section 2 can be generalized into a recursive framework for a better placement. The algorithm pursues a divide-and-conquer strategy to split the overall placement problem into a series of placement problems with smaller size (in terms of area for placement), such that the single-step framework can be executed recursively.

We assume that if a set of ANs $\{A_{\text{selected}}\}$ is obtained through the single-step framework, a better placement can be obtained from the neighbourhood of $\{A_{\text{selected}}\}$. Here, neighbors of $\{A_{\text{selected}}\}$ are located within $[x_{A_{\text{selected}}} \pm (\Delta x_A/2), y_{A_{\text{selected}}} \pm (\Delta y_A/2)]$, where Δx_A and Δy_A are the resolution of the original search space. In the recursive framework, based on the location of $\{A_{\text{selected}}\}$, a new set of candidates can be generated. Meanwhile, through the new set of AN candidates, a new set of ANs with a better placement can be found. Thus, the placement can be repeated recursively until the replacement of ANs cannot further reduce the operation cost in terms of average $\{r_i\}$, which can be simplified as $\|\{r_i\}_{\text{previous}} - \{r_i\}_{\text{current}}\|_2 / \|\{r_i\}_{\text{previous}}\|_2 \leq \varepsilon$, where ε is the predefined tolerance. In that case, selected AN candidates and their positions are declared as the final answer and the algorithm terminates. In summary, pseudocodes of the recursive placement framework are shown in Algorithm 1.

4. Performance Evaluation of the Framework

In this section, examples are used to show the performance of the framework. Computations run in an optimization solver Lingo 11 [19] on a 6 GB-RAM Intel i7 3.4 GHz PC.

4.1. Placement in a 2D Laboratory Environment. The performance of the system is analysed via a simulated example of a school laboratory with dimensions 50 m × 33 m [20–22], as shown in Figure 2. In order to monitor and regulate ventilations and thermal comfort of the laboratory, a CPS with sensors and actuators (i.e., electrical fans) is installed. A total 15 points of interest (POIs) and their consumed utilities are assigned according to positions of vents and pattern of
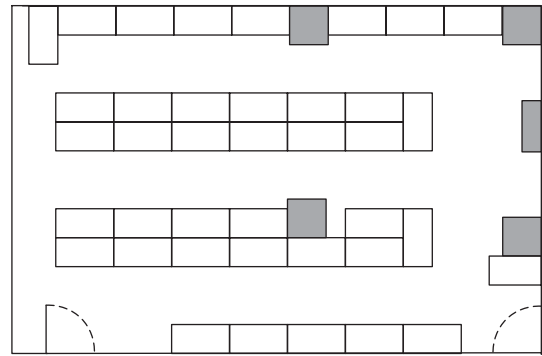

FIGURE 2: The floor-plan of the laboratory in the 2D field example. The grey squares and white rectangles denote wall structures and laboratory benches, respectively.

activities in the laboratory, as shown in Figure 3. In the example, service fading is excluded and single-step placement is used. Before the optimization, 100 candidates of ANs, which are located in areas bounded by outmost ANs, are generated. For every candidate, utility budget of an AN (C) is assumed to be 75 and $r_{\max} = 10$ m. After computations, the framework suggests that seven ANs with average $\{r_i\} = 5.785$ m are required to provide adequate requested services to all POIs. Results are shown in Figure 3(a). Figure shows that some ANs serve one critical POI and partially serve one noncritical POI, while others serve at most three noncritical POIs. Meanwhile, in some cases, one POI is served by two ANs. The problem contains 1036 variables (including 468 binary variables) and 4817 constraints, and the algorithm needs 30 minutes to obtain the optimal solution. Results with $r_{\max} = 1$ m are also shown in Figure 3(c). Figure 3(c) shows that if $r_{\max}$ is small, ANs work independently without sharing their duties.

4.2. Placement via a Recursive Framework. The recursive framework is then applied to the example in Section 4.1. In the example, 100 AN candidates have been used for optimization in the first iteration. 25 (5 × 5) candidates have been generated for each AN. Since seven ANs are needed for placement, totally 175 candidates have been used for optimization after the first iteration. In every iteration,
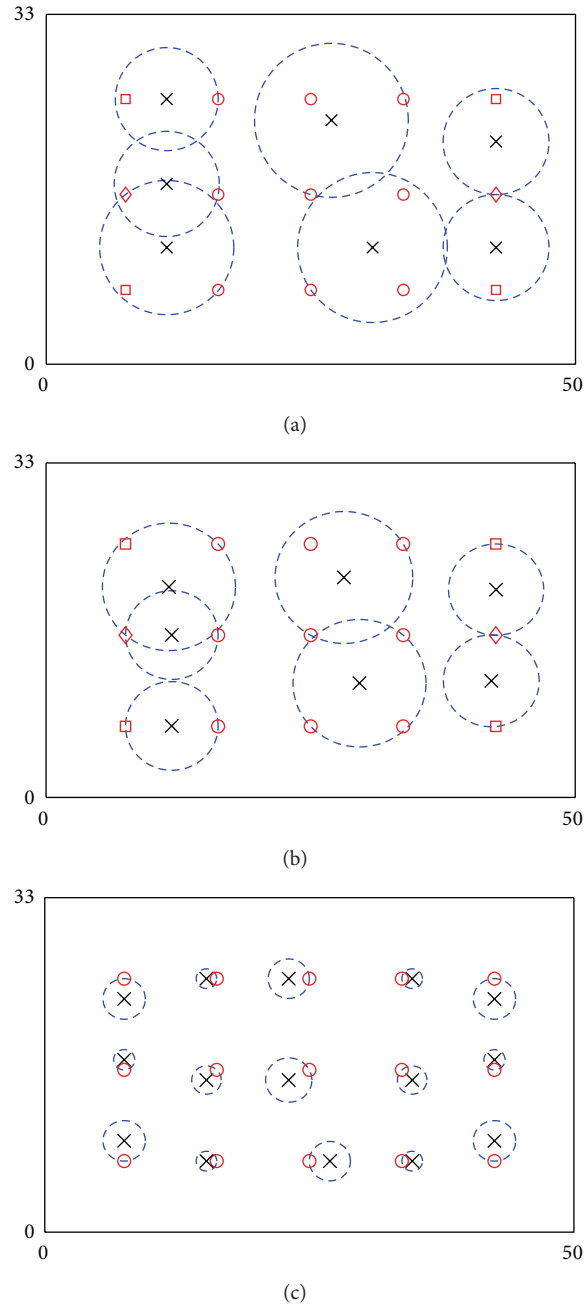
(a)

(b)

(c)

FIGURE 3: Locations of POIs and ANs in the 2D field example via the proposed framework: (a) optimization with r_{max} = 10 m, (b) optimization with r_{max} = 10 m after five iterations through recursive placement, and (c) optimization with r_{max} = 1 m. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares, and diamonds denote POIs with 25, 40, and 55 consumed utilities, respectively.

the algorithm terminates automatically after one hour if it cannot obtain an optimal solution. The obtained solution is then used to refine the search space for the next iteration. After five iterations, average coverage radius of ANs has been reduced by >8.98%. And the placement is shown in Figure 3(b). More details of results are shown in Table 1. Results show that recursive framework can improve the performance of ANs, with the expense of computation time.

4.3. Comparison with Other Strategies for Placements. Results of the proposed framework are compared with the following three strategies: (i) ANs are placed near the location of POIs one by one, with serving POIs with the highest utility consumption first and with utilization of residual utilities (Greedy approach, Strategy I), (ii) ANs are placed with considerations of geographical distribution of POIs only (Strategy II), and (iii) ANs are randomly placed with

TABLE 1: Results of the example via the recursive framework.

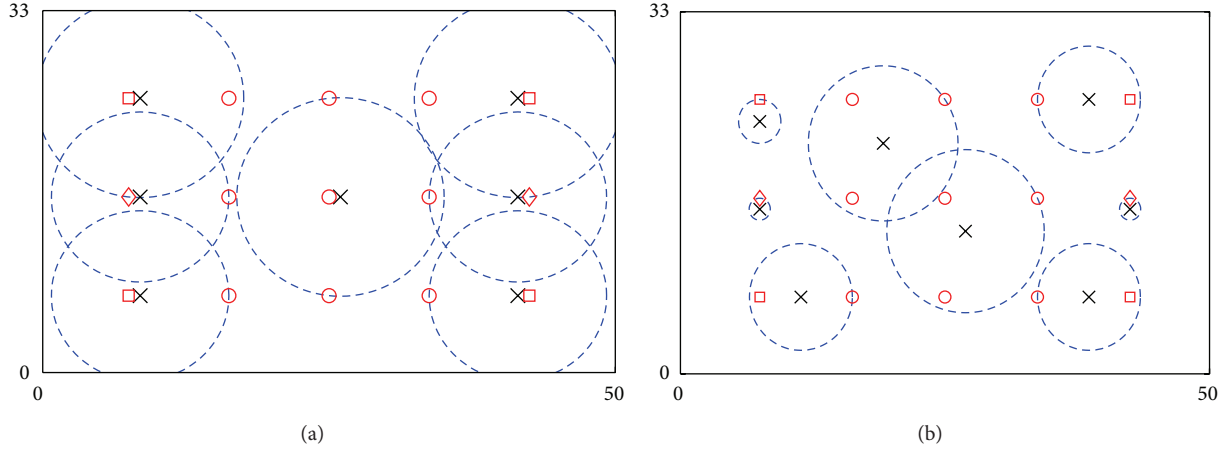| nth iteration	1	2	3	4	5
Average coverage radius (m)	5.7852	5.4523	5.2755	5.2706	5.2657

(a)   (b)

FIGURE 4: Locations of POIs and ANs in the 2D field example: (a) placement using Strategy I and (b) placement using Strategy II. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares and diamonds denote POIs with 25, 40, and 55 consumed utilities, respectively.

utilization of residual utilities (Strategy III). The example in Section 4.1 is used for evaluation. Results of the first two strategies are shown in Figure 4. Figures show that if Strategy I is used, seven ANs with average $\{r_i\}$ = 8.309 m are needed to serve all POIs. Meanwhile, if Strategy II is used, the system serves all POIs by eight ANs with average $\{r_i\}$ = 3.891 m. Furthermore, Strategy III has been tested ten times. Averagely, the system can serve all POIs by seven ANs with average $\{r_i\}$ = 11.546 m. Therefore, the proposed framework outperforms the other three strategies in terms of the number of ANs (i.e., system cost) or average $\{r_i\}$ (i.e., operation efficiency and system longevity). This is because the proposed framework considers geographic distribution of POIs and utilization of residual utilities for placements at the same time. These considerations become even more significant if the number of ANs is limited. These situations are typical because ANs are usually expensive to deploy.

4.4. Placements with Different Configurations.
Example in Section 4.1 is used for investigating the influence of various configurations. Results are shown in Table 2. From examples, we can arrive at a few conclusions about the result of placements as well as the efficiency of the framework.

(i) In general, the system requires fewer ANs as the utility budget (C) and maximum service coverage radius (r_{max}) of ANs increase. For example, only one AN is needed to serve the whole room if the actuator has an extremely large C and r_{max}. However, this assumption is usually not practical. Therefore, C and r_{max} should be configured carefully.

(ii) The space complexity of the optimization mainly depends on the number of POIs and number of AN candidates, while the time complexity mainly depends on the utility budget (C) and maximum service coverage radius (r_{max}). This is because there are usually more combinations of duty distributions between ANs for a larger C or r_{max}.

(iii) When the utility consumption is dominated by a few POIs or the utility budget of ANs is small, ANs tend to work independently without sharing job duty. On the other hand, when the utility consumption is evenly distributed, there is a higher possibility for ANs to share their workload. In this situation, the proposed framework can achieve a better performance.

(iv) If each AN can only serve at most two POIs due to limitations of hardware on ANs, through including constraint (12) for every ANs, the framework suggests that nine ANs with average $\{r_i\}$ = 4.238 m are needed to serve all POIs.

4.5. Impact of Fading on the Delivery of Services.
Example in Section 4.1 with different f and g = 0.01 is used for investigating the influence of service fading. Results are shown in Table 3. Results show that the number of required ANs increases as f decreases. It is because for a smaller f (i.e. a larger service power fading), POIs that are far from ANs consume more utility. In some situations, POIs that are too far from ANs cannot be served. As a result, additional ANs are added to serve these POIs. For example, if f = 1.5 m, workload cannot be shared among ANs; therefore, every POI is individually served by a nearby AN.

In the second example, an obstacle has been included, as shown in Figure 5. In the example, we assume the obstacle can effectively block the service and ANs can only serve POIs

TABLE 2: Results and computation complexity with respect to different problem configurations.

C	r_{max} (m)	Number of required ANs	Average $\{r_i\}$ (m)	CPU time (min.)
75	10	7	5.785	30
75	5	11	2.755	0.10
75	4.5	15	0.984	0.10
75	10	7	5.845	5
75	10	7	5.845	10
75	10	7	5.785	15
75	10	7	5.785	107 (Optimal)
50	10	10	5.288	60
25	10	20	2.5248	60
1000	25	1	21.865	0.58 (Optimal)

TABLE 3: Impact of service fading on the delivery of services with different f.

f (m)	Number of required ANs	Average coverage radius (m)	Average consumed utility (Scaled)	Average consumed utility (Original)
7.5	5	8.142	99.000	74.627
6.0	6	8.123	82.500	71.778
5.5	7	5.580	70.714	56.520
5.0	8	4.463	61.875	46.358
4.0	9	4.032	55.000	71.925
1.5	15	0.984	33.000	0.208

that lie in its line of sight. In order to model the obstacle, if the line of sight between A_i and P_j is blocked (i.e., the service is very difficult to be delivered from A_i to P_j), $f_{i,j} := 0.05$ m; otherwise, $f := 0.55$ m. Furthermore, some AN candidates are neglected since the obstacles have occupied spaces of some AN candidates. If the original placement is used, ANs are not able to serve some POIs. The new placement in Figure 5 shows that the system can still serve all POIs by placing ANs with similar configurations in different positions.

4.6. Multiscenario Placement. In this example, the system has to ensure the delivery of services in two scenarios: (i) the distribution of utility consumption in Section 4.1 is used as serving for office hours (Scenario A), and (ii) Figure 6(a) shows the distribution of serving for nonoffice hours (Scenario B). Figure 3(a) and Figure 6(a) show the placement of ANs if only Scenario A or B is considered, respectively. Meanwhile, Figure 6(b) shows the placement if both scenarios are considered for placements. The figure shows that 14 ANs with average $\{r_i\}$ = 5.174 m are needed to serve all POIs. Therefore, the framework can be used for multiscenario placements without introducing extra ANs.

4.7. Fault-Tolerant Placement with a Large-Scale Example. In order to demonstrate the fault-tolerant placement, the example in Section 4.1 with fault-tolerant parameters $K = 2$ and $K = 3$ is used. Placements of ANs for these two situations are shown in Figure 7. Figures show that nine ANs with average $\{r_i\}$ = 7.5841 m are necessary if each POI is served by at least two ANs (i.e., $K = 2$). Meanwhile, 14 ANs with average $\{r_i\}$ = 7.2357 m are necessary if POIs are served by at least
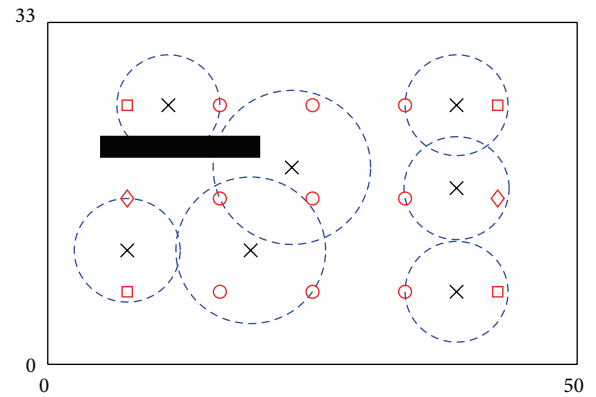

FIGURE 5: Locations of POIs and ANs in the 2D field example with the inclusion of an obstacle. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares, and diamonds denote POIs with 25, 40, and 55 consumed utilities, respectively. The black block denotes the obstacle.

three ANs (i.e., $K = 3$). Figures show that in order to ensure POIs can be partially served by ANs, only a few extra ANs are needed. Therefore, the algorithm allows ANs to provide fault-tolerant service through a cost-effective approach.

The second example is a large-scale example with 21 POIs. Their location and consumed utility are randomly assigned. A search space with 400 candidates has been generated for optimization. If $K = 2$, the framework requires 130 minutes to obtain the optimal solution, and the result is shown in Figure 8. The example illustrates that the proposed framework can solve large-scale placement problems.
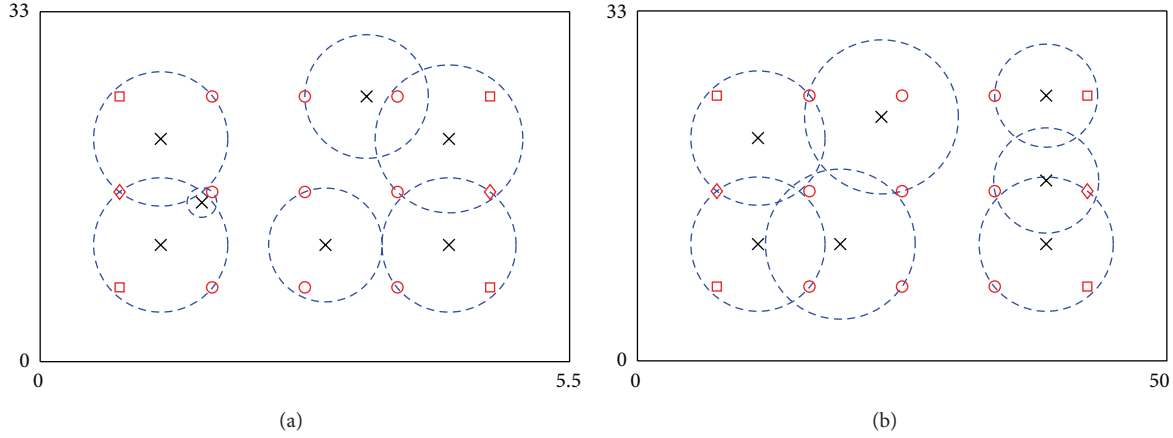
FIGURE 6: Locations of POIs and ANs in the 2D field example: (a) placement with Scenario B and (b) multiscenario placement with Scenarios A and B. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares, and diamonds denote POIs with 25, 40, and 55 consumed utilities in Scenario A and 35, 25, and 25 consumed utilities in Scenario B, respectively.
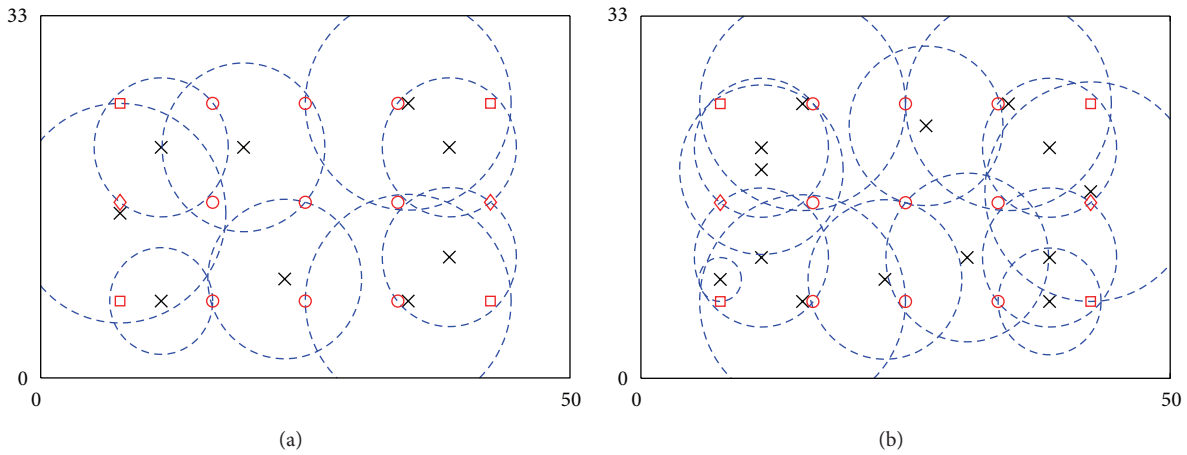

FIGURE 7: Locations of POIs and ANs in the 2D field example: (a) fault-tolerant placement with $K = 2$ and (b) fault-tolerant placement with $K = 3$. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares, and diamonds denote POIs with 25, 40, and 55 consumed utilities, respectively.

4.8. Placement in a 3D Field Environment. The performance of the framework is analysed via a simulated example of an arbitrary 3D environment with dimensions 80 m × 80 m × 80 m. A total of seven POIs are distributed randomly, with a random consumed utility ranging from 23 to 32. Totally, 512 candidates of ANs are distributed evenly in the 3D field. Through the optimization, six ANs with average $\{r_i\}$ = 7.594 m are assigned to provide adequate requested services for all POIs. The framework needs 33 seconds to solve the problem. The example proves that the proposed framework can be used for placements of ANs in 3D fields.

5. Conclusion

An actuator node placement framework with collaborative sharing of utility has been presented. In particular, partial serving and utilization of redundant residual utility in ANs are allowed in the relaxed formulation. Service fading, 3D placements, multiscenario placements, and fault-tolerant placement have been modeled in the framework. The problem has been converted into a binary integer linear programming optimization problem, such that the optimal solution can be obtained in a discretized search space.

We believe that this research is not only confined to schools, hospitals, and factory workshops, but also to many similar business environments such as offices. These environments are important to create positive benefits to the society as well as business activities. If we are to develop smart societies, we need to look at systems that are easily deployable, noninvasive, economical, effective, and energy efficient. Our algorithm for actuator node placement shows we can satisfy all these conditions.

In the future, we will generalize the framework for duty scheduling of ANs or robots in CPS, which aims at awaking several ANs or robots to work while putting others in sleep mode, such that the system longevity can be extended.
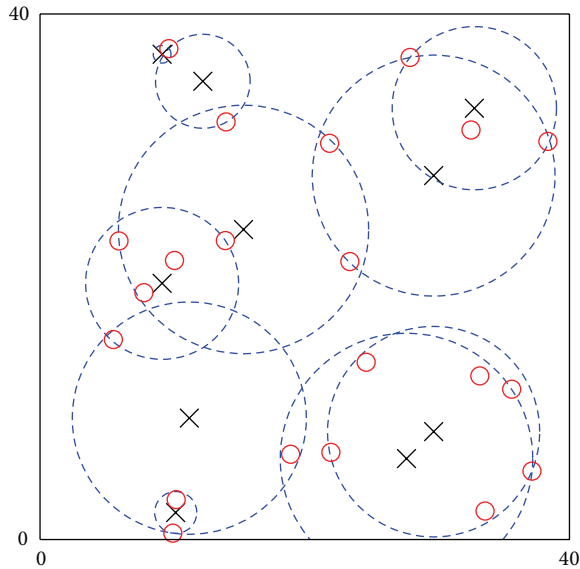
FIGURE 8: Placement of ANs in a large-scale example with $K = 2$. Crosses and dash circles denote ANs and service coverage of ANs. Circles, squares, and diamonds denote POIs with 25, 40, and 55 consumed utilities, respectively.

Further, to improve serving power, a smart power/battery management system will be potentially integrated into the framework.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.
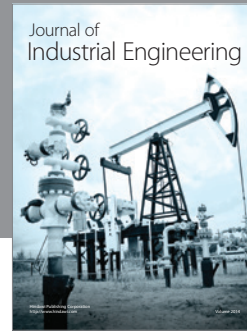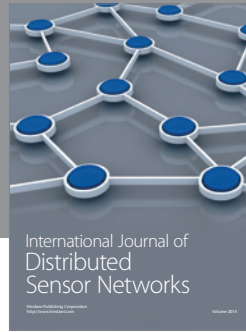
Acknowledgment

References

[1] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proceeding of the 47th Design Automation Conference (DAC '10)*, pp. 731–736, New York, NY, USA, June 2010.

[2] L. Parolini, B. Sinopoli, B. H. Krogh, and Z. Wang, "A cyber-physical systems approach to data center modeling and control for energy efficiency," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 254–268, 2012.

[3] I. Lee, O. Sokolsky, S. Chen et al., "Challenges and research directions in medical cyber-physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 75–90, 2012.

[4] K. Lakshmanan, D. de Niz, R. Rajkumar, and G. Moreno, "Resource allocation in distributed mixed-criticality cyber-physical systems," in *Proceedings of the 30th IEEE International Conference on Distributed Computing Systems (ICDCS '10)*, pp. 169–178, Genova, Italy, June 2010.

[5] J. C. Eidson, E. A. Lee, S. Matic, S. A. Seshia, and J. Zou, "Distributed real-time software for cyber-physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 45–59, 2012.

[6] P. Derler, E. A. Lee, and A. Sangiovanni Vincentelli, "Modeling cyber-physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 13–28, 2012.

[7] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: a survey," *Ad Hoc Networks*, vol. 6, no. 4, pp. 621–655, 2008.

[8] K. Akkaya and M. Younis, "C2AP: Coverage-aware and connectivity-constrained actor Positioning in wireless sensor and actor networks," in *Proceedings of the 27th IEEE International Performance Computing and Communications Conference (IPCCC '07)*, pp. 281–288, New Orleans, La, USA, April 2007.

[9] A. A. Abbasi, M. Younis, and K. Akkaya, "Movement-assisted connectivity restoration in wireless sensor and actor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 9, pp. 1366–1379, 2009.

[10] K. Akkaya, I. Guneydas, and A. Bicak, "Autonomous actor positioning in wireless sensor and actor networks using stable-matching," *International Journal of Parallel, Emergent and Distributed Systems*, vol. 25, no. 6, pp. 439–464, 2010.

[11] Y. Li, C. Vu, C. Ai, G. Chen, and Y. Zhao, "Transforming complete coverage algorithms to partial coverage algorithms for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 4, pp. 695–703, 2011.

[12] T. He, S. Krishnamurthy, J. A. Stankovic et al., "Energy-efficient surveillance system using wireless sensor networks," in *Proceedings of the 2nd International Conference on Mobile Systems, Applications and Services*, pp. 270–283, June 2004.

[13] C.-U. Lei, "LUOPAN: light utility-oriented placement of actuator nodes in sensor/actuator networks," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS 14)*, pp. 2563–2566, Melbourne, Australia, June 2014.

[14] C. H. Papadimitriou, "On the complexity of integer programming," *Journal of the Association for Computing Machinery*, vol. 28, no. 4, pp. 765–768, 1981.

[15] L. A. Wolsey, *Integer Programming*, John Wiley & Sons, New York, NY. USA, 1998.

[16] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[17] K. Ramachandran and B. Sikdar, "A population based approach to model the lifetime and energy distribution in battery constrained wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 576–586, 2010.

[18] S. Kellner, M. Pink, D. Meier, and E.-O. Blass, "Towards a realistic energy model for wireless sensor networks," in *Proceedings of the 5th IEEE Annual Conference on Wireless on Demand Network Systems and Services (WONS '08)*, pp. 97–100, Garmisch-Partenkirchen, Germany, January 2008.

[19] Official website of LINGO 14.0, http://www.lindo.com/products/lingo/.

[20] C.-U. Lei, H.-N. Liang, and K. L. Man, "Building a smart laboratory environment via a cyber-physical system," in *Proceedings of the IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE '13)*, pp. 243–246, 2013.

[21] C.-U. Lei, K. L. Man, H.-N. Liang, E. G. Lim, and K. Wan, "Building an intelligent laboratory environment via a cyber-physical system," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 109014, 9 pages, 2013.

[22] C. Lei, H. K.-H. So, E. Y. Lam, K. K. Wong, R. Y. Kwok, and C. K. Y. Chan, "Teaching introductory electrical engineering: project-based learning experience," in *Proceedings of the 1st IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE '12)*, pp. H1B–1–H1B–5, August 2012.