Simulation Model for Construction of Civil Works for Electrical Substations

Rania A. Fayed, A. Samer Ezeldin

Abstract

Electrical substations are considered by many construction experts as complex projects. The scope of civil works in electrical substations includes construction of several buildings/components divided into parallel and overlapped working phases that require variety of resources and are generally quite costly and consume a considerable amount of time. Therefore, construction of substations faces complicated resource scheduling problems, whereby the need to persistently discover approaches to enhance construction performance. To address the previously stated afflictions, this paper introduces a simulation model for the execution processes of civil works for typical electrical substations. The model is built on Anylogic software using discrete event simulation method and utilizes excel files for data entry. The main contributions of this model to the existent body of knowledge is to present at a macro level the scheduling and cost estimation of civil works for electrical substations while considering project variables and resource constraints. The proposed model provides the user with four main outputs with respect to the substation: i) resource loaded schedule, ii) cost estimation, iii) resources utilization breakdown and iv) amounts of materials consumed. The model is verified and validated on three different substations projects and a sensitivity analysis is conducted.

Keywords: Electrical substation, Construction, Simulation, Time, Cost, Resource

1. Introduction

An electrical substation is an auxiliary station of an electricity generation, transmission and distribution system where voltage is transformed from high to low or the reverse using transformers. It is utilized to switch generators, equipment, and circuits or lines all through a system. Additionally, it is utilized to change alternating current (AC) voltages starting with one level then onto the next, and/or change AC to direct current or direct current to AC.

The construction of electrical substation projects includes electromechanical works and civil works. Scheduling the execution processes for civil works in an electrical substation is a complex and challenging task. The works consists of multiple phases proceeding in parallel with each phase concerned with a group of buildings. All these buildings share resources (eg, equipment, labours and raw materials). The choice of resources is the most essential part of scheduling and ought to be considered in congruence with site limitations and the work to be attempted (Jaskowski & Sobotka, 2006).

Traditional project scheduling methods have limitations when used for these kinds of projects. The limitations are that they do not consider assigning resource accessibility limitations to project activities or the likelihood of time and cost savings through changing project schedules and resource modification (Kim & Ellis, 2010). The main problem is that these traditional methods focus on nearby optimality at activity level and not on global optimality at project level (Ashuri & Tavakolan, 2015).

To surmount these restrictions, several simulation and modelling techniques have been proposed to allow for the computer scheduling of construction projects. The upside of these techniques is that they consider the possibility of
interfering with project activities and moving shared resources crosswise over project activities to simultaneously optimize total project duration, total project cost, and aggregate variety of resource allocation (Zheng, Ng, & Kumaraswamy, 2005, Kim & Ellis, 2010, Ashuri & Tavakolan, 2015).

This investigation presents a simulation model to schedule the execution process of civil works of electrical substation using Anylogic software based on database excel file for input data entry to estimate the time and cost of the substation. The simulation model shows the real-time sequence of events that the project would run through to finish all activities of the substation using shared resources. This is accomplished while addressing their availability as a constraint, maintaining different relationships between the activities, and taking into account the variable parameters. Moreover, the model estimates the cost of the substation including direct and indirect costs, percentage of utilization of each resource (equipment and labours) and consumed amount of each material.

2. Main Components of Electrical Substations

Electrical substations are unique projects. They, typically, consist of the same buildings and components with often the same design. Sometimes the areas of the buildings and number of components differ from one substation to another according to the total area of the substation and the electrical power the substation should deliver. The scope of civil works includes all construction, finishing and sanitary works. The main civil work components of an electrical substation are (Figure 1):

- Gas insulated switchgear (GIS) building, this building is designed to house the gas insulated switchgear, auxiliary equipment, distribution panels, cables and raceways, bus/breaker arrangement, maintenance area and the portion serviced by a crane;
- Control building, this building is designed to house medium voltage switchgear, auxiliary transformers, batteries, power supply system, telecommunication system, and control and protection system;
- Switchyard portals foundations, these are concrete footings designed to carry out the steel structure of various apparatus for all switchyard electromechanical equipment;
- Transformers foundations, these are concrete foundations of two types; foundations for outdoor switchyard power system transformers and foundations for switchgear power system transformers;
- Fire walls, these walls are protection reinforced concrete (R.C.) walls constructed between every two transformers foundations;
- Underground R.C. fire footings, they are constructed for outside galvanized steel firefighting water tank;
- Firefighting buildings, they are R.C. buildings with suitable area constructed to maintain firefighting system for transformers provided with all necessary equipment and accessories;
- Workshop building, it is a R.C. building constructed for storage, maintenance and workshop works;
- Cable trenches, R.C. trenches are required for installation of buried cables, conduits, piping, etc.
- Galvanized steel fence that surrounds the outdoor switchyard and the outdoor transformers area and rested on concrete footings;
- Roads are required for operation and maintenance of the substation and to get easy access to any part inside the substation. Roads are divided into main roads and secondary roads;
- Shed roof for vehicles, sun shed parking for vehicles is arranged, so that, one shed for buses and trucks and the second shed for cars;
- Ground and elevated tank, water is stored in the ground tank and water supply is provided with water pressure maintained by an elevated tank;
- Drainage tanks, those are underground R.C. tanks (septic tank, filter tank and collecting tank) connected with manholes constructed for the drainage of sewage water.

It is difficult to manage a project including all of the above mentioned buildings/ components using traditional methods. These methods are ineffective in actualizing multiple resources sharing procedure as it deals predominantly with activities' durations not with resources which might bring about large resource fluctuations. Moreover, the traditional methods focus in dealing with changing project schedules and resource modification on nearby optimality at activity level. Therefore, managing the schedule of such projects needs a procedure which is
able to imitate the real construction operation over time and deal with the concept of multiple resource allocation and resource sharing between activities according to resources availability as well as time on global optimality at project level to obtain accurate results and better project performance.

Figure 1. Typical plan of electrical substation

3. Simulation Modeling for Construction Scheduling

Simulation is an analytical technique involving designing and experimenting with a built up mathematical-logical model. At the point when relationships between variables are not linear and/or arbitrary variables are incorporated in the problem, simulation is the best apparatus to show the relationships and solve the problem. Simulation enables schedulers to think about the practices of a procedure without the need of detailing a mathematical function of an unpredictable behavior input. In other words, if the problem is too complex for mathematical formulation, it is best analyzed by simulation (Srisuwanrat, 2009).

Kavanagh (1985) created SIREN (Simulation of Repetitive Networks) computer model for repetitive construction projects. It simulates the different crews as they line to complete activities. A working schedule and cumulative cost curve are created and statistics are accumulated on crew and equipment utilization, all being yielded graphically. A Monte-Carlo simulation is additionally included as probability distributions might be connected with the duration of each activity. However, the model has some limitations. The model is based on activity on node method in scheduling and it doesn’t consider resource pool concept during simulation which is ineffective in dealing with resource allocation. The model doesn’t take into account project parameters such as production rates, number of resources available and changing the number of resources assigned to each activity, while it depends on entering the mean duration of activities and equipment utilization as input data which might obtain inaccurate results. Moreover, the model doesn’t allow the user to impose his priorities and plan of work on the system.

Lutz, Halpin, & Wilson (1994) presented the system used to model the learning advancement marvel in the CYCLONE format utilizing the Boeing learning curve. The learning development improvement was coded in the MicroCYCLONE (a microcomputer version of CYCLONE) environment using QuickBASIC programming. The upgrade permits the user to determine the rate of learning expressed as a percentage, the acknowledgment tally
required for the task duration to increase because of learning, and the limit at which no additionally learning change can be figured out. However, significant improvement in individual processes did not always result in a significant improvement for the overall operation.

Chehayeb & AbouRizk (1998) added to a way to deal with empower powerful utilization of systems simulation in the scheduling of construction projects. The methodology was actualized in a product program named SimCon, which upgrades current simulation techniques using continuous production links for activities that contain a project. The continuous production links give construction managers an adaptable representation of activity sequencing. The methodology expands on existing CYCLONE fundamentals through new builds to interface forms in a simulation-based project plan. Illustrations from an actual construction project demonstrate the upsides of the continuous production links. However, the model didn’t address the resource allocation or resource sharing problem and its effect on time and cost.

Isidore & Back (2002) introduced another strategy called the multiple-simulation analysis technique (MSAT) to address the problem of relating the probabilistic cost estimate and project schedule data such that high percentile values for both of these instruments could be chosen that were connected in some significant way. MSAT consolidates discrete event simulation, regression analysis, and numerical analysis with a specific end goal to build up a model that clarifies the relationship between the stochastic cost estimate and the schedule data. But the model concentrates only on the time-cost relationship, without including the effect of resource factor on both time and cost.

Peña-Mora, Han, Lee, & Park (2008) presented a hybrid simulation model that consolidates system dynamics (SD) and discrete event simulation (DES) which have basically been used to analyze the strategic and operational issues separately, for infrastructure construction projects. The simulation model is applied on a non-typical repetitive earthmoving process as an example. The simulation results show that a precise coordination of strategic viewpoint and operational points of interest is useful to upgrade the procedure performance by empowering construction managers to recognize potential procedure improvement areas that customary methodologies might miss. Although, the model strategic objective is to finish the project with minimum operation cost and time, however, to achieve this objective the model is based on an assumption of unlimited resource availability (management actions) which does not actualize the real operation. Also, the study does not extensively consider side effects or ripple effects caused by these management actions.

Labban, AbouRizk, Haddad, & Elsersy (2013) developed a special purpose simulation model to estimate, plan and manage asphalt paving operations, and quickly portray a proposed alternative strategy currently being research to rapidly assemble construction simulation models. The model framework proposes recognizing a typical arrangement of information structures that will convey product, process and ecological data, adding to an algorithm that can order that information into suitable inputs for simulation, and building up a generic discrete event simulation engine capable of self-replication and handling of the input data to create a typical DES history of the model. However, the proposed methodology is still a theoretical approach, in its early stages, which needs to be developed, implemented and evaluated. This approach needs further development of the conceptual model, classification of the appropriate target user group(s), developing the modular input data structure components and identifying the appropriate simulation environment in which to apply it.

Zankoula & Khoury (2014) created a generic and dynamic Discrete Event simulation model utilizing AnyLogic programming that can be embraced by contractors with a specific end goal to plan and optimize the construction process of on-shore wind farms in Lebanon and elsewhere. The created work delineates the distinctive construction stages from rough grading, access roads construction, foundation and electrical works, to wind tower assembly and erection. The entire procedure is then optimized to essentially minimize the project duration. But, the model doesn’t consider the resources’ production rates in calculating activities’ durations and also it doesn’t include cost information.

Zankoul, Khoury, & Awwad (2015) created two simulation models, a Discrete-Event Simulation (DES) model and an Agent-Based Simulation (ABS) one, using the multi-method simulation software AnyLogic 7.1 to address work focused at proficiently overseeing logistics of earthwork operations in the construction field, in particular cut
and fill processes, and hauling activities. Both models represent efficiency in describing earthwork construction process, however the models doesn’t include important project parameters such as cost information and resources’ production rates.

From the previous review and evaluation of the existing simulation models, it can be noticed that the approaches lack a combination of significant parameters in the same developed model such as appropriate modeling language, time-cost-resource tradeoff representation, production rates, resource constraints and resource allocation. Therefore, the construction industry is in need for a simulation model with an adapting modeling language that can imitate the real construction operation over time and deal effectively with resource allocation and resource sharing concepts. Such a model should also consider the project parameters and resources constraints to aid the decision makers in planning, scheduling, control, resource allocation and cost estimation of such construction projects.

4. Methodology for Model Development

The simulation model for electrical substations is developed using Anylogic software and utilizes a data base excel file for input data entry. Basic software platform (BSP) has been utilized in simulation for simplicity of model building, improved visual model environment, as well as adaptability and animation. The interesting adaptability of the BSP modeling language empowers the user to capture the multifaceted nature and heterogeneity of systems to any craved level of detail. BSP’s graphical interface, apparatuses, and library objects permit user to rapidly show differing ranges, for example, assembling and logistics, business processes, resources, and suppliers’ conduct. The object-oriented model outline worldview bolstered by BSP accommodates particular, hierarchical, and incremental development of expansive models.

BSP bolsters the consistent combination of discrete and continuous simulations. The native Java environment bolsters boundless extensibility including custom Java code, outside libraries, and outer information sources. A broad statistical distribution function set gives an incredible stage to mimicking the instability natural in all systems. Furthermore, the powerful experimental framework offers a wide assortment of simulation approaches.

BSP includes a graphical modeling language i.e. the model is built in a graphical editor that allows the user to edit the diagram of the model graphically. The graphical model elements are called “Active objects”, they are the main building blocks of BSP models. Active objects can be used to model very diverse objects of the real world. These blocks are shown in a palette view grouped by categories in a number of stencils (palettes) and libraries.

The substation model is built using discrete event simulation method. Discrete-Event simulation modeling likewise, alluded to as “Process-Centric” in BSP, comprises of isolating a consistent procedure into discrete parts to simplify analysis. The procedure being referred to is displayed as a sequence of discrete events or activities. Each event happens at a specific moment in time and denotes a change of state in the framework. Between sequential events, no adjustment in the framework is accepted to happen; therefore the simulation can specifically hop in time from one event to the next. Entities arrive into the model at a specific arrival rate and go through its activities each one in turn consecutively. The BSP’s Process Modelling Library is utilized to build this model as it supports discrete-event, or process-centric modeling paradigm.

The development of this model is built depending on using the capabilities of Anylogic software mentioned above and adapting the software for scheduling and cost estimation of the electrical substation model considering the project variables and resource constraints in java language. So, the formulation and computational equations of the model are done using java coding through over the model as will be discussed later.

The execution process of civil works of the substation depends on dividing the total construction area into three zones (phases) working in parallel to each other as they are not related to or dependent on each other. Therefore, the model is divided into three zones; each zone includes a group of buildings which are related to the same construction area as following:

Zone 1: This zone is related to gas insulated switchgear area which includes GIS building and control building.
Zone 2: This zone is related to switchyard area which includes portals foundation, transformers foundations, fire footings, fire walls, chain link fences, cable trenches, main roads and secondary roads.

Zone 3: This zone is related to service buildings area which includes firefighting buildings, workshop, ground and elevated tank, drainage tank and vehicles shed.

Building the model goes through (10) main stages as follows:

- Creating a database excel file;
- Creating a new model called “OzModel”;
- Creating a java class called “Util”;
- Creating the flow charts diagrams of Zone 1 model- agent type;
- Creating the flow charts diagrams of Zone 2 model- agent type;
- Creating the flow charts diagrams of Zone 3 model- agent type;
- Cost Estimation of the substation including direct and indirect cost;
- Exporting start and finish dates of activities to represent them on a bar chart;
- Creating the “Simulation” experiment;
- Model verification using three case studies.

5. Stages for Model Development

5.1 Creating a Database Excel File

To facilitate the data entry to the model, a Microsoft Excel file is used, so that the user can easily enter or change all input data needed to operate the model. The database file consists of two sheets. Sheet 1 (Figure 2) includes the bill of quantities of all clauses of the substation, categorized by type of building, a resource breakdown for each clause; manpower, equipment and materials, unit cost for each raw material and indirect cost. Sheet 2 (Figure 3) in database excel file includes the production rates for resources (equipment and manpower crews), the maximum number available from each type of resource for each zone in the model, the number required from each type of resource for each building and the unit cost of each resource per day.

The model includes 28 types of resources (equipment and manpower crews), and 88 types of raw materials. During the operation of the model, the resource constraints are the maximum number available from each type of resource for each zone in the model. Moreover, the model takes into consideration the following project variables affecting time and cost: 1) the production rates for resources (equipment and manpower crews); 2) the number required from each type of resource for each building; 3) unit cost of each resource per day; 4) unit cost for each raw material; and 5) indirect unit cost including fixed and variable cost. The utilization of this data in the model is described in the following sections.

5.2 Creating a New Model “OzModel”

A new model shell is created within Anylogic called “OzModel”.

5.3 Creating a Java Class “Util”

A java class called “Util” is subsequently created within Anylogic in order to write all java codes used to set the type of resource pools and number of resource units needed from each resource pool to be seized by activities in order to be executed, each code will be called by the active object (activity) which utilizes this type of resource.

5.4 Creating “Zone 1” Model- Agent Type

As mentioned previously, Zone 1 is the model representing switchgear area in the substation which includes GIS Building and Control Building. Zone 1 model graphical interface represents the flow charts for each of these buildings, resource pools used by active objects, parameters indicating the number of resource units seized by each active object from each resource pool, all types of raw materials utilized by active objects represented each in a material queue storage, parameters indicating unit cost and total cost of each resource pool, parameters indicating
the quantity utilized, unit cost and total cost of each raw material, parameters indicating indirect cost and all java codes need to calculate these values. After completing the process, the graphical editor will be opened to build the model.

**Figure 2. Part of data base sheet 1 excel file**

<table>
<thead>
<tr>
<th>ID</th>
<th>Work Description</th>
<th>Unit</th>
<th>Quantity</th>
<th>Excavation</th>
<th>Disposal</th>
<th>Sand</th>
<th>Aggregate</th>
<th>Louder</th>
<th>Compactor</th>
<th>Plain Concrete</th>
</tr>
</thead>
<tbody>
<tr>
<td>109</td>
<td>Excavation</td>
<td>m3</td>
<td>3013</td>
<td>m3</td>
<td>3013</td>
<td>3013</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>Replacement layer</td>
<td>m3</td>
<td>1005</td>
<td>m3</td>
<td>7839</td>
<td>5427</td>
<td>1005</td>
<td>1005</td>
<td></td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>Plain concrete for foundation</td>
<td>m3</td>
<td>449</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>Wood for plain concrete for foundation</td>
<td>m2</td>
<td>75</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>113</td>
<td>Reinforced concrete for foundation and necks of columns</td>
<td>m3</td>
<td>267</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3. Part of data base sheet 2 excel file**

5.4.1 **Database**

The first step is to make the model read input data from the database excel file and this is done by dragging the Excel File element from the Connectivity palette onto the graphical diagram. In the Properties view, the name of the element is modified to “DB” in the Name field. Excel file in which this object will work with is specified in “File” field. In order to read any cell value from “DB” excel file, a java code is utilized in the place where this cell is called.

5.4.2 **Resource Pools**

The equipment and manpower crews are defined in the model as resource pools. A resource pool defines a set of resource units that can be seized and released by entities using flowchart objects, in which the same resources are shared between activities and move inside the project as entities. In the Properties view (see Anylogic manual), the name of the element is modified to the name of resource in the “Name” field. The type of resource is chosen from “Resource type” drop box, either Moving, Static or Portable. The capacity of each resource means the maximum number of units available from this resource. The capacity is defined using “Capacity defined” drop box methods. The resource pool capacity is read from the database file using java code.

5.4.3 **Raw Materials**

Each type of raw materials is represented in the model by two objects: Source object and Queue object, connected together with a connector. Source object is used to start generating of raw material, while Queue object is used to store the raw material in the specified order until it is utilized by active objects. In the Properties view of Source object, arrival method of raw material is chosen from “Arrivals defined by” check box to be Rate and the rate is defined in “Arrival rate” field by units per day. In the Properties view of Queue object, capacity is defined in “Capacity” field by using java code.
5.4.4 Resource Parameters

The number of resource units needed by each active object (activity) from each resource pool in order to be executed is defined as a parameter. The values of these parameters are read from the database excel file. Resource parameters are created for each building in the zone. In the Properties view of Parameter object, the type of resource unit is chosen from the “Type” field drop box to be integer number (int). The number of resource units needed is defined in the “Default value” field. This number is read from the database excel file using java code.

5.4.5 Flowcharts

A Flowchart is the sequence of work for each building represented by active objects connected together by connectors indicating the relationship between objects. Zone1 consists of two flowcharts; GIS building flowchart and control building flowchart. The active objects used to build these flowcharts from process modeling library are source, split, service, assembler, combine, select output, batch and sink.

The Source object is used to generate entities; it is usually a starting point of a process model. In the Properties view of Source object (see Anylogic manual), method of arrival of entities is chosen from “Arrivals defined by” checkbox to be “Rate”. Arrival rate is defined in “Arrival rate” field as number of entities arrived per model time unit which is day. If there are multiple entities arrive in the same time, “Multiple entities per arrival” box is checked. If a specific number of entities will arrive, the “Limited number of arrivals” box is checked, and the number of entities is defined in “Maximum number of arrivals” field.

The Service object is used to describe the execution process of an activity. It seizes a given number of resource units, delays the entity, and releases the seized units. In the Properties view of Service object (see Anylogic manual), two options are introduced to seize resources, in the seize field, either resources just of one single type, so “units of the same pool” option is chosen, or resources of different types, so “(alternative) resource sets” option is chosen. The type of resources and the number of resource units (resource parameters) to be seized for this service are defined in “Resource sets” field by calling the activity resource set code from “Util” java class and specifying the resource parameters and types of resource pools. The time this service will take to be executed (\(D.T\)) is defined in “Delay time” field by using the following equation:

\[
D.T. = \frac{Q}{P \times N}
\]  

where “\(Q\)” is the activity quantity, “\(P\)” is the resource production rate and “\(N\)” is the number of resource units (resource parameter).

All fields of the equation are read from database excel file by using java code. The quantity of raw materials required for this service is called from each raw material queue by using a java code written in “On enter” field. This code reads the required quantity of each raw material from database excel file and calls it from the raw material queue store.

For every approaching entity (“original”), the Split object makes one or a few different entities and yields them by means of via outCopy port. A new entity created may have an importance of a copy, a sibling, etc. and can be of subjective sort. The number of new entities is changed dynamically. This is used to make the entity pass in several parallel paths at the same time.

The Assembler object permits certain number of entities from a few sources (5 or less) to be joined into a solitary entity. The type of the new entity, and in addition its instatement, is determined in the new object. The quantity of entities for each input required to deliver one new entity is additionally determined utilizing the object parameters (Quantity 1, Quantity 2, etc.). All arrived entities hold up inside the object until all required entities arrive. Once the new entity can be build, the assembly operation starts. This operation takes the time specified in the Delay time field. In the Properties view of Assembler object, the quantity of entities required to arrive from each input to create the new entity is specified in “Quantity 1, Quantity 2, Quantity 3, Quantity 4 and Quantity 5” fields. Seize resources, resource sets, delay time and quantities of raw materials are set following the same steps as mentioned previously in Service object.
The Combine object waits for two entities to arrive (in subjective request) at ports in1 and in2, produces another entity and yields it. The new entity might be a “totally new”, i.e. a recently built object whose properties potentially rely on the original entities, or it might be one of the original entities. Once the two entities are prepared, the combine operation takes zero time.

The Select Output object routes the approaching entities to one of the two output ports relying upon (probabilistic or deterministic) condition. The condition may rely on the entities and on any outside variables. The entities invest zero time in Select Output. In the Properties view of Select Output object, there are two options introduced for “Select True output” field, either the entities are routed randomly with specified probability, in this case “With specified probability [0..1]” is chosen, and the probability of true route is defined in “Probability” field. The other option routes the entities depending on specified boolean condition, in this case “If condition is true” field is chosen, and the condition is defined in “Condition” field using java code.

The Batch object changes over various entities into one entity by either disposing the original entities or making another one (permanent batch), or by adding the original entities to the contents of the new entity (temporary batch). This object implants a Queue object where it stores the approaching entities. Once the number of the stored entities achieves the batch size, the batch is made and leaves the object instantly.

Finally, the Sink object disposes entities. It is usually an end point in a process model.

Figure 4 shows control building flowchart. All buildings flowcharts are built following the same template according to the activities of each building and relationships between activities.

5.5 Creating “Zone 2” and “Zone 3” Models-Agent Types

Creating Zone 2 and Zone 3 model-agent types is performed utilizing the same steps of creating Zone 1 as illustrated in section 5.4. Zone 2 model includes eleven flowcharts; each flowchart represents a building /component in the switchyard area. These components are: 1) portals foundations type F1, 2) portals foundations type F2, 3) transformers foundations 220 k.v., 4) transformers foundations 66 k.v., 5) fire footings, 6) fire walls, 7) chain link fences, 8) cable trenches inside switchyard, 9) cable trenches outside switchyard, 10) main roads and 11) secondary roads. While Zone 3 model includes five flowcharts that represent five buildings/components in the service buildings area. These components are 1) firefighting buildings, 2) workshop, 3) elevated tank, 4) drainage tank and 5) vehicles shed.

5.6 Cost Estimation

The Cost estimating component of the model consists of direct cost and indirect cost and each one includes fixed costs and variable costs. Direct cost is the cost of materials, equipment and manpower crews, while indirect cost is the cost of salaries, overheads and other site expenses. Fixed cost is a constant value which does not change by project duration variation. While variable cost is a function of time, it varies according to the total project duration. Direct fixed cost is the cost of raw materials and direct variable cost is the cost of equipment and manpower crews. Indirect fixed cost includes site offices, mobilization, taxes and deductions by Employer. Indirect variable cost includes staff wages, head office overhead and site facilities (flats rent, cars, water, electricity, phones, site expenses and etc.).

5.6.1 Estimation of Direct Variable Cost (Resource Pools)

Equipment and manpower crews are defined in the model as resource pools. Estimating the cost of a resource unit in a resource pool is done by creating a custom resource type to obtain the utilization of a unit as a percent of time the unit was busy during simulation. Knowing it, the total number of days each unit was in work (D) can be calculated as follows:

\[ D = T \times U \]  \hspace{1cm} (2)

where “T” is the total model time units (days) and “U” is the percentage of utilization of each resource unit.
Figure 4: Control Building Flowchart
After multiplying it by the cost of one unit per day, total cost is obtained.

The specific steps are as follows:

First, Custom resource type is created for each resource pool by dragging the *Agent element* from the General Library palette into the graphical editor. In the new agent wizard, the type of agents to be created is specified as population of agents (resource units) and the name of the agent is defined to be the name of the equivalent resource pool. In the properties view of the agent object, the agent is set to be used in the flowchart as resource unit. The properties of each resource pool object are adjusted so that the resource units are defined in the model by the custom resource type equivalent to this resource pool type.

Then, two variables are created for each resource pool; the first variable is created to specify the value of unit cost of this resource per day, this cost value is read from database excel file using java code. The second variable is created to determine the value of total cost of this resource pool at the end of model simulation or at any time the simulation is terminated. Creating a variable is done by dragging the *Variable object* from the General Library palette into the graphical editor. In the properties view, the value of the variable is specified in “Initial value” field. After creating the custom resource type for each resource pool, adjusting resource pool properties and creating cost variables, each resource pool cost is calculated using a java code written in “on destroy” field, in the properties view of each Zone-Agent type. The aim of the code is to estimate the total time units of simulation model, the percentage of utilization of each resource in the resource pool, the time units this resource was utilized by multiplying the percentage of utilization by the total model time units, the cost of the resource by multiplying the unit cost of the resource per day by utilized time units, the total cost of all units in the resource pool and finally the total cost of all resource pools.

5.6.2 Estimation of Direct Fixed Cost (Raw Materials)

The cost of each raw material is determined by first identifying the amount of material that has been withdrawn from the storage (queue of this material) and then multiplying this amount by the unit cost of this material which is read from the data base excel file to obtain the total cost of each material. These specific steps are as follows:

In order to determine the utilized quantity of each material, a variable is created for each material and then increased by increment 1 each time an active object withdraw this material from the queue to add this amount to the variable. This is done by first dragging the Variable object from the General Library palette into the graphical editor. In the properties view, the name of the variable is specified in the “Name” field. The type of the variable is specified in the “Type” field to be integer number (int). The initial value of the variable is specified in “Initial value” field to be zero. In “On enter” field of the active object properties, where the quantity of each material is called to be utilized, the variable name is called using java code in order to add this utilized quantity in the material variable (for example, in each time, calling sand material quantity, the following code is written “usedSand++”).

Then, to read the unit cost of each material from the database excel file, a variable is created for each material. In the properties view, the type of the variable is specified to be double and the initial value of the variable is specified using java code to read the unit cost of this material from the excel file.

The Estimation of the total material cost in each zone is obtained by multiplying the unit cost variable of each material by the total consumed quantity variable of this material. This equation is written using java code in “on destroy” field, in the properties view of each Zone-Agent type.

5.6.3 Estimation of Indirect Fixed Cost

To set the indirect fixed cost in the model, a variable is created to read this value from the database excel file using java code.
5.6.4 Estimation of Indirect Variable Cost

The indirect variable cost is defined in the model as a variable indicating cost per unit time (day), and the total indirect variable cost \((T.I.V.C)\) is estimated at the end of the model run by multiplying the indirect variable cost per day \((I.V.C.D)\) by the total model time days \((T)\). A variable is created with type double to read this value (indirect variable cost/day) from the database excel file using java code. The total indirect variable cost is determined by writing the following equation using java code in “on destroy” field, in the properties view of each Zone-Agent type:

\[ T.I.V.C = I.V.C.D \times T \]  \[3\]

5.6.5 Estimation of Total Cost

Total cost of each zone is estimated by adding the total direct fixed cost (materials), the total direct variable cost (resource pools), the total indirect fixed cost and the total indirect variable cost. This equation is written using java code in “on destroy” field, in the properties view of each Zone-Agent type.

5.7 Export Activities’ Dates

In order to view the model on a bar chart schedule, the start and finish dates for each activity in the model should be exported to an external excel file in order to later represent them on a bar chart using a scheduling software package like Primavera or Microsoft project. The function object and the java code which is written in the properties view of each active object (activity) are utilized to export activities’ dates (see Anylogic software).

6. Model Verification

The simulation model’s operation is verified by applying the model on three substations (case studies) of different quantities. The three substations are located in Egypt. The substations have different sizes and transmit different powers. In order to test the stability and accuracy of the output results of the model, sensitivity analysis is applied to test the linearity of the results. This is done by fixing all the input data during running the model for each of the three substations, while changing only the bill of quantities for each substation. The constant input data includes unit cost of resources, indirect cost, production rates, maximum available resources and number of resources needed by each building. These constant input data are the actual data of Elnabq substation.

6.1 Description of Substations (Case Studies)

The substations (case studies) that are used for model verification are Elnabq substation, New-Esna substation and North-Qena substation. The data of these case studies are shown in Table 1. Table 2 shows the components of each substation and the number and total area of each component. The utilized case studies consist of the main components described in “Main Components of Electrical Substations” section.

6.2 Results and Evaluation

The first step to run the model is to enter the input data in the database Excel file as mentioned previously in section 5.1. For each substation, the simulation model is run for each zone based on the input data to obtain the total model time units and cost. Figure 5 shows the total time units for each zone in each substation and the total model time units for each substation. While Figure 6 shows the total direct cost for each zone in each substation and the indirect cost and total cost of each substation.

It can be noticed from Figure 5 that the total time of all zones of New Esna substation is more than Elnabq substation. So, for Zone 1, time increased by 213 days than Elnabq substation model time, for Zone 2, time increased by 96 days, while for Zone 3, time increased by 25 days. And the total model time increased by 213 days. This returns to the reason that the quantities of activities of New Esna substation are greater than Elnabq substation, in addition, all the other input parameters which control time are fixed in both substations. These parameters are production rates, maximum available resources and number of resources required for each building. So, the time increased linearly according to the increase in quantities.
On the other hand, it is found that the total time of all zones of North Qena substation is less than Elnabq substation. So, for Zone 1, total time decreased slightly than that in Elnabq substation by 16 days because the quantities of buildings in Zone1 (GIS building and control building) are slightly less than those of Elnabq substation. While, for Zone 2, it is found that the total time of this zone in North Qena substation decreased greatly than Elnabq substation by 163 days and this because North Qena substation doesn’t transmit power of 220 k.v., so it doesn’t include all the construction components that carry 220 k.v. equipment, which are all located in switchyard area that is represented in Zone 2 model, such as, portals foundation, transformers foundation for 220 k.v., cable trenches inside switchyard. Besides, the decrease in quantities of the other components due to the reduction in land area between both substations. Also for Zone 3, the model time in North Qena substation decreased by 66 days than Elnabq substation due to the reduction in quantities.

Moreover, due to the variation in time and quantities between the three substations, the cost also changes. It is observed from Figure 6 that the cost of all zones in New Esna substation is more than Elnabq substation. So, the direct cost increased by 1238211.3 L.E., 1611092.45 L.E. and 269651.74 L.E. for Zone1, Zone 2 and Zone 3 respectively. And this returns to the increase in quantities and time due to the fixation of resources constraints in both substations. The indirect cost increased by 1251600.71 L.E. This increase due to increase in time, and as the indirect variable cost is function of time, it will increase linearly with time.

On contrary, the cost of North Qena substation decreased than Elnabq substation due to the decrease in quantities and total model time. So, the direct cost decreased by 621370.87 L.E., 2794818.4 L.E. and 314016.11 L.E. for Zone1, Zone 2 and Zone 3 respectively. This reduction in cost due to inexistence of the components specific for 220 k.v. power equipment and the reduction of the quantities of the existing components. Also the indirect cost decreased by 93866.72 L.E. due to the reduction of total model time units of North Qena substation.

Finally, it is observed that the model output results of time and cost for the three substations change linearly by changing the quantities of activities for each substation due to keeping all other input parameters constant, which are production rates, maximum available resources, resources needed by each building, unit cost of resources and

---

**Table 1. Data of Case Studies Substations**

<table>
<thead>
<tr>
<th>Location (city)</th>
<th>Power (k.v.)</th>
<th>Land Area (m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elnabq Substation</td>
<td>220/66/11</td>
<td>552.15</td>
</tr>
<tr>
<td>New Esna Substation</td>
<td>220/66/11</td>
<td>789.74</td>
</tr>
<tr>
<td>North Qena Substation</td>
<td>66/11</td>
<td>752.22</td>
</tr>
</tbody>
</table>

**Table 2. Description of Components of Case Studies Substations**

<table>
<thead>
<tr>
<th>Components</th>
<th>Elnabq substation</th>
<th>New-Esna substation</th>
<th>North-Qena substation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>Total area (m²)</td>
<td>Number</td>
<td>Total area (m²)</td>
</tr>
<tr>
<td>GIS building</td>
<td>1</td>
<td>1321</td>
<td>1</td>
</tr>
<tr>
<td>Control building</td>
<td>1</td>
<td>934</td>
<td>1</td>
</tr>
<tr>
<td>Portals foundations F1</td>
<td>211</td>
<td>646.19</td>
<td>250</td>
</tr>
<tr>
<td>Portals foundations F2</td>
<td>25</td>
<td>617.18</td>
<td>37</td>
</tr>
<tr>
<td>Transformers foundations 220 k.v.</td>
<td>2</td>
<td>196.8</td>
<td>2</td>
</tr>
<tr>
<td>Transformers foundations 66 k.v.</td>
<td>4</td>
<td>201.84</td>
<td>4</td>
</tr>
<tr>
<td>Fire walls</td>
<td>2</td>
<td>37.52</td>
<td>2</td>
</tr>
<tr>
<td>Fire footings</td>
<td>2</td>
<td>19.98</td>
<td>2</td>
</tr>
<tr>
<td>Chain link fences</td>
<td>2</td>
<td>385</td>
<td>2</td>
</tr>
<tr>
<td>Cable trenches inside switchyard</td>
<td>1</td>
<td>626.2</td>
<td>1</td>
</tr>
<tr>
<td>Cable trenches outside switchyard</td>
<td>1</td>
<td>155.8</td>
<td>1</td>
</tr>
<tr>
<td>Main roads</td>
<td>1</td>
<td>552.15</td>
<td>1</td>
</tr>
<tr>
<td>Secondary roads</td>
<td>1</td>
<td>522</td>
<td>1</td>
</tr>
<tr>
<td>Firefighting buildings</td>
<td>2</td>
<td>71.5</td>
<td>2</td>
</tr>
<tr>
<td>Workshop building</td>
<td>1</td>
<td>76.37</td>
<td>1</td>
</tr>
<tr>
<td>Ground and elevated tank building</td>
<td>1</td>
<td>129.96</td>
<td>1</td>
</tr>
<tr>
<td>Drainage tank</td>
<td>1</td>
<td>68.59</td>
<td>1</td>
</tr>
<tr>
<td>Vehicles Shed</td>
<td>1</td>
<td>120.88</td>
<td>1</td>
</tr>
</tbody>
</table>

On the other hand, it is found that the total time of all zones of North Qena substation is less than Elnabq substation. So, for Zone 1, total time decreased slightly than that in Elnabq substation by 16 days because the quantities of buildings in Zone1 (GIS building and control building) are slightly less than those of Elnabq substation. While, for Zone 2, it is found that the total time of this zone in North Qena substation decreased greatly than Elnabq substation by 163 days and this because North Qena substation doesn’t transmit power of 220 k.v., so it doesn’t include all the construction components that carry 220 k.v. equipment, which are all located in switchyard area that is represented in Zone 2 model, such as, portals foundation, transformers foundation for 220 k.v., cable trenches inside switchyard. Besides, the decrease in quantities of the other components due to the reduction in land area between both substations. Also for Zone 3, the model time in North Qena substation decreased by 66 days than Elnabq substation due to the reduction in quantities.
indirect cost. Also, the results of Elnabq substation show close agreement to the real case study due to utilizing the actual data of the substation as input data. And this proves the stability of model operation and accuracy of output results.

Figure 5. Comparison between Model Time Units Results for the Three Substations

![Figure 5](image.png)

Figure 6. Comparison between Model Cost Results for the Three Substations

![Figure 6](image.png)

After running the simulation model for Elnabq substation, the start and finish dates of each object (activity) in the model are exported into an external excel file “logDB” as described in “Export Activities’ Dates” section; these dates are represented on a bar chart schedule using Primavera software (Figure 7).

7. Conclusion

The primary purpose of this research is to develop a simulation model that shows at a macro level scheduling the execution process of civil works of electrical substation construction projects considering all project variables and constraints. The model is designated to aid decision makers in planning, scheduling, control and cost estimation of the civil works of substation projects. The model is built by discrete event simulation method using Anylogic software based on a database excel file for input data entry. The model can be considered as a template for the execution process of civil works of any substation project. The model is verified by applying it on three substations of different quantities and powers. A sensitivity analysis is applied to test the accuracy of output results. Moreover,
the model shows efficiency by providing time and cost results for Elhabq substation in close agreement to the real case study. Also, the sensitivity analysis shows linearity of output results for the three substations, which proves stability of model operation.

![Bar Chart Schedule of Elhabq Substation Using Primavera Software](image)

Figure 7. Bar Chart Schedule of Elhabq Substation Using Primavera Software

The simulation model presented in this research provides the user with four main outputs with respect to a new electrical substation project: (1) a resource loaded schedule according to the specified resource constraints by the user; (2) cost estimation for the project including direct and indirect cost; (3) resources utilization breakdown including equipment and manpower crews representing number of working days and cost for each resource unit; and (4) amounts of materials consumed through the project duration.

REFERENCES


