A Configurable Approach for Modelling and Verifying Smart Grids

1st Thang Bui
Ho Chi Minh City University of Technology (HCMUT), Vietnam National University - Ho Chi Minh City
Ho Chi Minh City, Vietnam
bhthang@hcmut.edu.vn

2nd Tuan Bui
Ho Chi Minh City University of Technology (HCMUT), Vietnam National University - Ho Chi Minh City
Ho Chi Minh City, Vietnam
tuanbc88@hcmut.edu.vn

3rd Tuyen Nguyen
Ho Chi Minh City University of Technology (HCMUT), Vietnam National University - Ho Chi Minh City
Ho Chi Minh City, Vietnam
ndtuyen@hcmut.edu.vn

4th Anh Nguyen
Ho Chi Minh City University of Technology (HCMUT), Vietnam National University - Ho Chi Minh City
Ho Chi Minh City, Vietnam
duyanhnguyen@hcmut.edu.vn

5th Liem Nguyen
Ho Chi Minh City University of Technology (HCMUT), Vietnam National University - Ho Chi Minh City
Ho Chi Minh City, Vietnam
nvliem@hcmut.edu.vn

6th Prof.Liem's friend
XXX University
Somewhere in the world
someone@somewhere

***Abstract*—Research on smart grids, specially on modelling and verifying their capacities such as detecting, reacting, pro-acting and self-healing, has been carried out decades ago. Whilst most of the modelling approaches are conventional based as they focused on the grid topology, the verification therefore suffers from the state space explosion, the verification main drawback.**

In this research, our new approach to represent smart grids using Colored Petri Net (CPN) will be represented. The proposed approach allows engineers to re-configure the system easily to adapt to any change in the grid without re-modelling the system from the grid topology. Our experimentation has shown that the the state spaces of the nets constructed by the new approach are smaller than that of the conventional modelling approach. This opened a great opportunity to the verification for smart grid to overcome its 'inherently intractable' drawback. The proposed verification method and experimentation in this work confirmed our hypothesis.

***Index Terms*—smart grid modelling, configurable modelling, smart grid verification.**

I. Introduction

Smart grid is a grid power system using information and communication technology to optimize the transmission and distribution of electricity between producers and consumers and consolidate the electricity infrastructure with the inter-information infrastructure. Smart grids allow pro-actively detecting and reacting to changes in usage and multiple issues of the electricity system, and they have self-healing capabilities.

Studies of smart grid have been carried out decades ago and spans from equipment studies, systems, communications, optimization, network security ... In general, the research methodologies of those studies are almost folded into two concerns: (1) modelling approach and (2) simulation/testing/verification. Although formal verification is the most noticeable method for verifying the correctness, it requires that the system has to be specified formally and it suffers from the well-known state space explosion problem in practice.

Recent studies have shown that properties like load balancing or the probability of a fault occurring in the grid (and fault resilience) can possibly be tested by the formal methods [1]–[3]. However, these studies also show that the effectiveness of formal verification (test time, computer resources used, and scalability) depends on how the system is modeled and which formal verification technique is used.

The most notable studies on modeling and testing the system model have been studied in [4]–[8]. These studies focus on modeling wireless sensor networks and using abstraction techniques, clustering, etc., applying model testing algorithms to verify congestion characteristics of wireless sensor networks.

In addition, the research on improving the effectiveness of model checking, a "more-practical" formal method, has also been studied in [9]–[13] with abstract methods, heuristic search, randomization, parallelization, indexing ... Studies on verification of smart grids in Vietnam seemingly have not been published.

This research contributes the following three-folds: (1) smart grid representation using Color Petri nets [14] that allows engineers to re-configure the model (of a grid) for any changes in the original smart grid without re-modelling it; (2) a quick review on comparison of the new approach to conventional modelling approach; (3) an heuristic search approach for finding solutions proving "smart" capacities of the grids.

The rest of this paper is organized as follows. In the next section, background information, including the review on conventional modelling approach for smart grids, is presented. The new formal representation approach for smart grids, including a comparison to the conventional approaches, is in section III. A proposed heuristic search for verification smart grid capacities in abnormal situations is in section IV. The experimentation is then presented in section V. The last section is for conclusion and future works.

II. Background

A. Smart grids

A smart grid is an electrical grid/network that empowers the control of smart devices based on data collected from the network to respond immediately to demands for electricity. It allows to improve the electrical network in reliability, availability and efficiency[1].

During operation, the smart grid can be listed by level such as: the customer, the distribution system, and the transmission system. At the level of the customer, it may include some things like meters that can be read automatically, meters that communicate to customers, control of customers' loads, and flexibility in the use of time-of-day or time-of-use meters. At the level of the distribution, system may include distribution system automation, selective load control, managing distributed generation and "islanding". And the level of the transmission, system may include measurement of phase and other advanced measurements, and other advanced control devices, distributed and autonomous control [15].

In summary, the goal of smart grid designing is to provide grid observability, create controllability of assets, enhance power system performance and security, and reduce costs of operations, maintenance, and system planning. In the research of smart grid, there are some computational tools for modeling and analysis of the smart grid [16]–[21], as well as studies on threats and solutions [22].

B. Formal verification

Formal verification is a research direction aimed to prove the correctness of a system with respect to a certain property represented in a formal specification. Theoretically, the verification is a process of finding a formal proof on a formal/mathematical model of the system under test. There are two main approaches in formal verification: model checking and theorem proving.

C. Colored Petri Nets

A Petri net is a weighted, directed graph of nodes as places and transitions, to model systems. Directed arcs join places to transitions (as inputs) and transitions to places (as outputs). The Petri networks by transmitting tokens from (input) places to transitions and from transitions to (output) places (firing). The amount of tokens transmitting is defined by the weight of directed arcs.

[1]https://smartgrid.gov

Colored Petri net (CPN) [14] is an extension of Petri net which combines the strengths of Petri nets with the expressive power of functional programming languages by adding more features and properties to tokens, places, and transitions resulting in more classes of high level. It allows tokens to have a data value attached to them called the token color, and each place represents a color set. Moreover, arc-expressions (an extended version of arc weights in classical Petri nets) specify which tokens can flow over the arcs. Additional guard constraints on enabling transitions can also be defined as Boolean expressions.

A CPN is a tuple $CPN = (C; B; P; T; V; F)$, where

- C is a finite set of color sets.
- B is a bag of tokens (value) of colors $c \in C$.
- P is a finite set of places.
- T is a finite set of transitions.
- V is a finite set of variables $v \in V$.
- F is a finite set of functions.

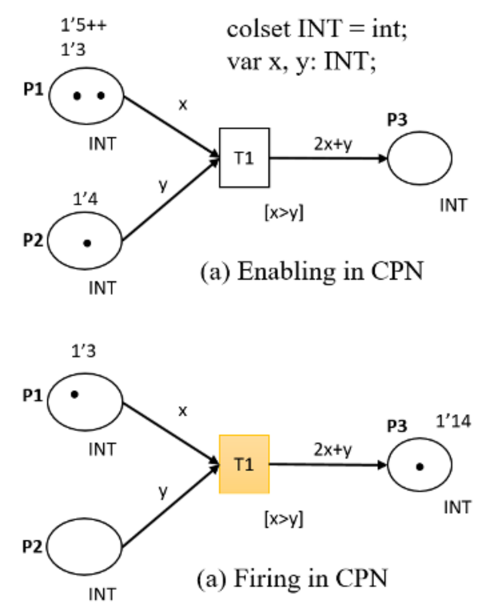

Fig. 1. Colored Petri Net (CPN).

For example, Fig. 1 shows a CPN in its two stages: enabling and firing. In Fig. 1(a), place $P1$ contains 2 bags of integer tokens (one with 5 tokens and another with 3 tokens) and place $P2$ contains a bag of 4 integer tokens. They are the input places for transition $T1$ with the guard expression $x > y$ (from $P1$ and $P2$, respectively) as the condition to enable the transition. In In Fig. 1(b), the transition $T1$ has been fired when the guard expression is confirmed (the bag of 5 tokens in $P1$ and the bag of 4 tokens in $P2$). It took (those) tokens from place $P1$ and $P2$ and generated tokens in $P3$ (a bag with 14 integer tokens as the result of the output arc-expression $2x + y$).

D. Conventional modelling approach review

1) Smart grid representation using Simple Petri nets: In this paper, we use a smart grid whose topology depicted in Fig. 2, as a demonstration case study. This smart grid consists of two generators (denoted as filled circles), three consumers (denoted as double-lined circles), one battery (denoted as hexagon), two buses (hollow circles) and seven circuit breakers (CB - squares). These elements will be called nodes in the grid.

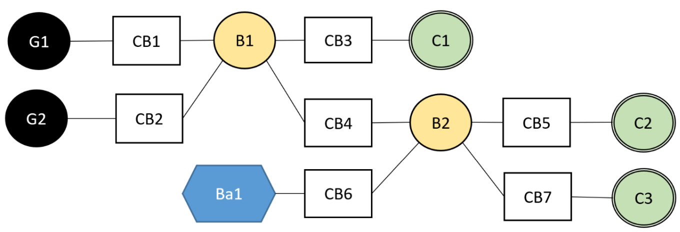

Fig. 2. Topology of a sample smart grid.

It is assumed that all elements in a smart grid have standard parameters such as id and type. Moreover, some of them such as generators and consumers have additional parameters such as capacity for their generating capacity or consuming capacity. Table I is for the parameters and a configuration of the sample grid.

TABLE I
A SMART GRID CONFIGURATION

Item	Id	Type	Capacity
Generator 1	1	GEN	2
Generator 2	2	GEN	1
Battery 1	3	BAT	3
Consumer 1	4	CON	3
Consumer 2	5	CON	1
Consumer 3	6	CON	2

2) Smart Grid Topology representation: Previous studies on modelling and verification of smart grids using Petri nets or Colour Petri nets have been studied decade ago [1]–[3], [16]–[20]. They are all try to represent the net as similar as possible to the network topology, and apply some verification techniques in checking some desired properties.

For example, in [18], all generators are defined as places in a Petri net. It is good for engineers to simulate the nets as they are familiar with the topologies. However, for large networks, the Petri nets are too complicated to be represented and simulated and are too hard to be upgraded. Even though the work in [20] represents grids in CPN, the models are still large when it focuses on describing detail/local electrical transformer areas to detect and localize illegal loads.

In general, the conventional modelling approach represents smart grid elements as petri net elements (places and transactions) separately. By using this approach, the grid in Fig. 2 can be represented in CPN as in Fig. 3. It is easy to see that, the net is similar to the original topology of the grid.

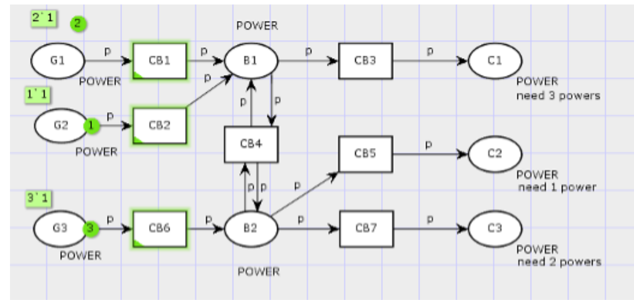

Fig. 3. Conventional approach for modelling the sample smart grid in Fig. 2.

3) State space analysis: The state space of a model system is a set of possible configurations that system can have. For example, it is a set of possible states of the grid in Fig. 2. In this work, we use the integrated State Space module in CPN tool[2] to analyze state space and Graphviz library[3] for exporting state space to a .dot file, then create a graph from the .dot file.

The state space of the grid in Fig. 2 is illustrated in Fig. 4. It is very complicated as it consists of 300 states and 1000 arcs for a very simple smart grid. Based on the state space, a suitable configuration for fulfilling the consumers are $<CB1: ON, CB2 : ON, CB3 : OFF, CB4 : ON>$. It is a short description of a path from the initial state to the final state in the state space $<N1, N2, N5, N5, N32, N64, N105, N161, N213, N258, N277, N295, N299>$ (the blue line, 13 steps long).

III. CONFIGURABLE MODELLING APPROACH

A. Smart grid representation using Color Petri nets

Upon suffering on the huge and complex of the state space of the conventional modelling approach, we proposed a new re-configurable modelling approach for smart grids based on CPN. In this approach, the grid can be represented using two generic components: generation and transmission/ consumption. Each of these components is modeled by a CPN. An illustration of those representations is in Fig. 5.

Fig. 5(a) is a pure petri net created from the topology in Fig. 2 by reducing the busses and CBs. The left part of the petri net describes the power generation with three Generators G1, G2 and G3. From these three generators, power is generated through transitions “gen1”, “gen2” and “gen3” and is stored in the place “Generated”. After being generated, power will be transferred and consumed in place “Consumer 1”, “Consumer 2” and “Consumer 3” through transitions “trans1”, “trans2” and “trans3”, respectively. To leverage the power of the Color Petri net for creating a simpler net, all generators and consumers are folded into the places Generator and Consumer, respectively, as shown in Fig. 5(b). All transitions gen1, …trans1, …have also been folded to functions “fn_gen”, “fn_trans” and “fn_cons” as in Listing 3

[2]CPN tools. Available: http://cpntools.org
[3]https://graphviz.org/

Fig. 4. The state space of the sample smart grid using conventional modelling approach.

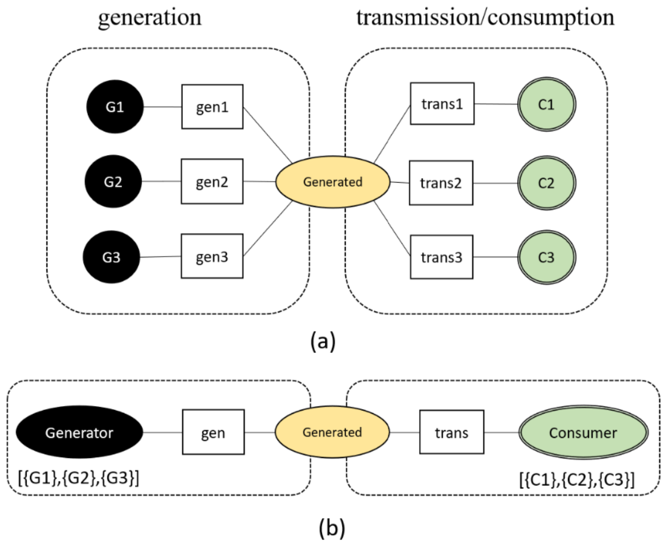

Fig. 5. Two components: generation and transmission (a); folded generation and folded transmission (b).

Listing 1
DECLARATIONS OF THE COLOR SETS.

```
colset IDX = int;
colset TYPE = with GEN|CON|BAT;
colset CAPACITY = int;
colset POWER = int;

colset NODE = record i:IDX*t:TYPE*c:CAPACITY;
colset NODE_POWER = product NODE*POWER;
```

to describe power generation, transmission and consumption, respectively. The detail of how to represent the components is in the section III-B.

The declarations are presented using the CPN-ML syntax of the CPN tool as follows (see Listing 1).

Color set types:

- "IDX": integer, a unique id of a node in the grid.
- "TYPE": enumeration = ["GEN", "CON"] for Generator, and Consumer (Load).
- "CAPACITY": integer, the capacity of a generator or consuming capacity of a consumer.
- "NODE": a record of IDX * TYPE * CAPACITY, a node in the grid.
- "POWER": an amount of power.
- "`NODE_POWER`": a product of "NODE" and "POWER", for power generated or consumed by the NODE.

Configuration:

The configuration of the grid described in Table I is in Listing 2. It is also called the initial marking of the CPN.

Arc-expression:

Listing 2
INITIAL MARKING OF THE "GENERATOR" AND "CONSUMER" PLACES.

```
val initGenerator: NODE_POWER =
    [({i=1,t=GEN,c=2},2)
    ,({i=2,t=GEN,c=1},1)
    ,({i=3,t=GEN,c=3},3)]
    ;
val initConsumer: NODE_POWER =
    [({i=4,t=CON,c=3},0)
    ,({i=5,t=CON,c=1},0)
    ,({i=6,t=CON,c=2},0)]
    ;
val initGenerated: POWER = 1`0;
```

Listing 3
FOLDING FUNCTIONS USED IN THE NEW APPROACH.

```
fun fn_gen(p1: POWER,p2: POWER) = p1+ p2;
fun fn_cons(n: NODE, p: POWER) =(n,p+(#c(n)));
fun fn_trans(n:NODE, p:POWER) = p-(#c(n));
```

The folding functions used as arc-expression in firing CPN are defined as in Listing 3.

B. Smart grid component representation

In this section, we describe more about the components mentioned above. The generation component has two places "Generators", "Powers" and a transition "gen". The Generators place contains all generators of the smart grid. For example, the Generator place in the left of Fig. 5(b) contains three generators G1, G2 and G3 of the smart grid in Fig. 2. For example, the Generator G1 (id=1, maximum power generation capacity=2) is setup to generate power of 2 (100% of maximum capacity) is represented as $(\{i = 1, t = GEN, c = 2\}, 2)$. With this approach, we can easily simulate a stable or unstable power source as well as the on/off state of the power source.

In this component, the "gen" transition fires when taking one NODE_POWER token from the place "Generator", which represents a generator and its associated power generation, and a POWER token from the place "Generated", which represents the total amount of electricity that has been produced and stored. The function "fn_gen" will sum the amount of electricity available from the generator and the amount of electricity available in the "Generated" place to create a new total.

Fig. 6 depicts the state of the grid implemented on the CPN Tool after the transition "gen" fires once upon receiving G1 tokens $(\{i = 1, t = GEN, c = 2\}, 2)$ (from the Generator place) and 0 power (the token 1'0 in the Generated place) to generate 2 powers (the new token 1'2 in the Generated place and and no more token of G1 in the Generator place).

After being successfully generated, the electric power is ready to be transmitted and consumed using the transmission

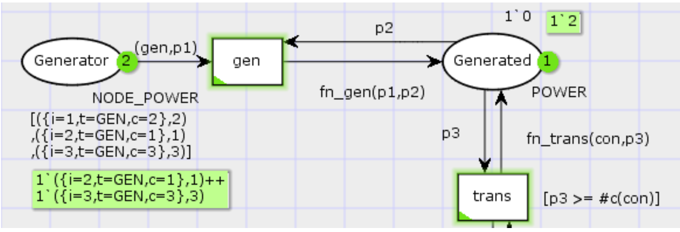

Fig. 6. The implementation of the generator component on CPN tool.

component. The "Consumer" place in the right of Fig. 5(b) contains all consumers of the grid with initial configuration.

Transition "trans" fires when receiving one token from place "Generated" and one from place "Consumer". The function "fn_trans" and "fun_cons" are used for transferring and consuming power between two places "Generated" and "Consumer" Fig. 7 describes a state of the grid when the place "Consumer 3" received 2 powers (the token is changed to $1`(i = 6, t = CON, c = 2, 2)$).

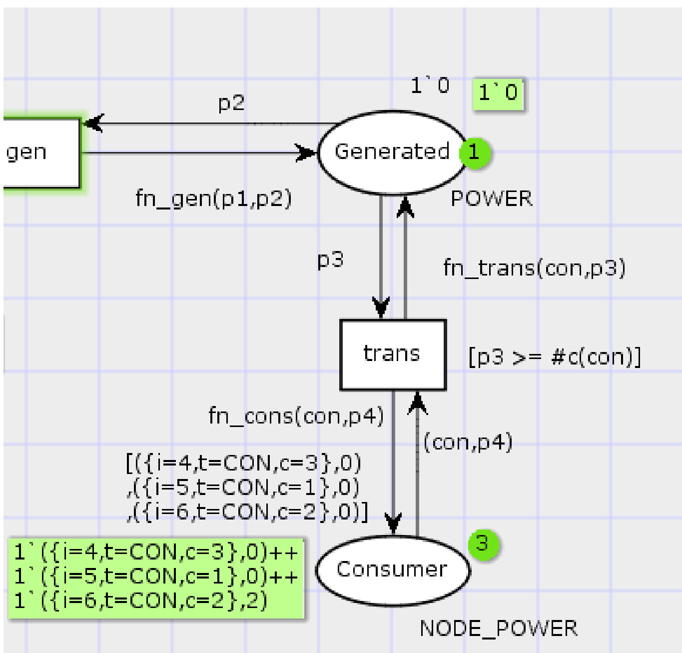

Fig. 7. The implementation of the transmission component on CPN tool.

The full model of the demonstration smart grid is illustrated in Fig. 8. It can be seen that no matter what the complexity of the smart grid, the presentation model in CPN is small.

C. State space analysis

The state space of the grid in Fig. 2 modelling using our new approach is illustrated in Fig. 9. It only consists of 71 states and 150 arcs and can be analysed manual. Based on the state space, a suitable configuration for fulfilling the consumers are $<CB1 : ON, CB2 : ON, CB3 : OFF, CB4 : ON>$, a short description of a path from the initial state to the final state in the state space $<N1, N3, N9, N18, N34, N51, N61>$ (the blue line, 7 steps long).

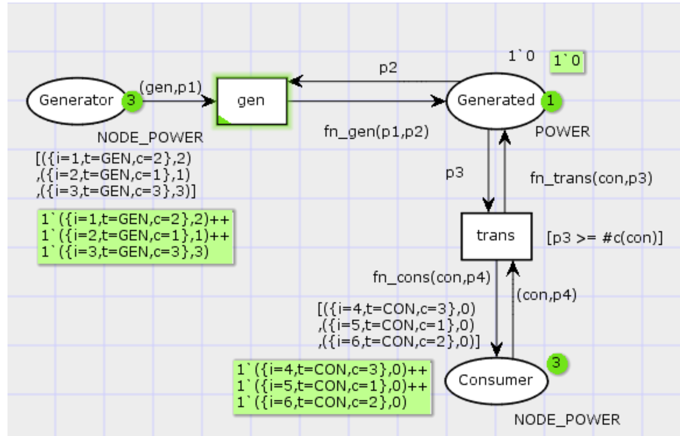

Fig. 8. The implementation of the smart grid on CPN tool.

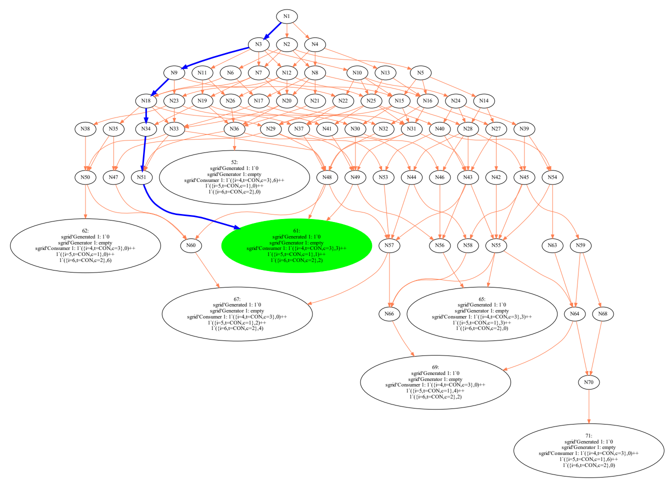

Fig. 9. The state space of the sample smart grid using new modelling approach.

The state space in this case is very simple compared to that of the conventional approach. It is much smaller in two-aspects: (1) number of arcs or the complexity of the state space: 150 arcs compared to 1000 arcs; and (2) the length of the path of finding a suitable working configuration for the grid: 7 steps compared to 13 steps.

D. Configurability

To adapt to the changes in topology, the conventional approach asks the engineers to re-model the grid (by removing or adding some net components and by updating the firing rules). In our new approach, one needs only to re-configure the arc-expressions.

For example, when there is no more Generator 1, the only thing to be changed is the variable "initGenerator" to exclude out the color set for the Generator 1. When a new consumer comes into the smart grid, the model is still the same, but the variable "initCosumer" is updated include a new color set for the new consumer. Of course, for changes in element properties such as capacities, the CPN model can be updated easily by changing only the parameters of related elements. Those changes can be programmed easily.

So, in general, the model of the smart grid can be easily be updated accordingly to the changes in topology and properties of the smart grid elements without re-modelling the grid.

E. A comparison of the two approaches

In this section, we try to compare the smart grid state spaces generated between the conventional approach and the usage of our proposed approach. Table 3 shows the configuration of the grid in the other six case studies. We increase some of the power of these Generators to show an explosion of state space and running time.

In case 01 and case 02, the total amount of power generated is equal to the total amount of power consumed. In case 03 and case 04, the total amount of power generated is greater than the total amount of power consumed. And vice versa for case 05 and case 06.

TABLE II
THE CONFIGURATION OF THE GRID IN FOUR CASE STUDIES.

Case Studies	**G1**	**G2**	**G3**	**C1**	**C2**	**C3**
Case Study 01	2	1	3	3	1	2
Case Study 02	10	5	6	6	9	6
Case Study 03	30	30	30	30	10	20
Case Study 04	100	50	60	30	10	20
Case Study 05	20	10	10	20	10	20
Case Study 06	10	10	10	20	10	20

F. Experimental results

The experimentation results are represented in Table VIII for the conventional approach and Table IX for our new approach.

For the conventional approach (see Table VIII), the CPN tool can analyze the full state space only when the power capacities of Generators are small as in the first two cases in seasonable amount of time (~ 0 second and 149 seconds). However, when the power capacities of Generators increased, as in case 03, case 04, case 05 and case 06, the CPN tool can only analyze a part of the state space and took more time (over 300 seconds).

For our new modelling approach (see Table IX), the CPN tool can analyze the full state space for all six cases very fast (almost zero seconds). The generated state spaces are also extremely smaller (up to 50,000 times) compare the that of the conventional approach.

The comparisons of state spaces of the two approaches is illustrated in Fig. 10.

IV. AN HEURISTIC SEARCH FOR VERIFICATION OF SMART GRIDS

For enabling self-healing capacity of smart grids, the configuration of (smart devices on) the grids in abnormal situations have to be studied before hand. By using our modelling approach, the exploration of grid situations can become the search on the state space.

TABLE III
THE RESULTS OF FOUR CASE STUDIES IN CONVENTIONAL APPROACH.

Case	**Nodes**	**Arcs**	**Run time (sec.)**	**Status**
Case 01	300	1 000	~ 0	Full state space
Case 02	47 124	240 828	149	Full state space
Case 03	41 685	194 744	over 300	Partial state space
Case 04	40 724	189 766	over 300	Partial state space
Case 05	41 225	195 409	over 300	Partial state space
Case 06	40 423	192 794	over 300	Partial state space

TABLE IV
THE RESULTS OF FOUR CASE STUDIES IN NEW MODELLING APPROACH.

Case	**Nodes**	**Arcs**	**Run time (sec.)**	**Status**
Case 01	71	150	~ 0	Full state space
Case 02	48	96	~ 0	Full state space
Case 03	144	332	~ 0	Full state space
Case 04	974	2 780	~ 0	Full state space
Case 05	38	72	~ 0	Full state space
Case 06	26	46	~ 0	Full state space

In this work, we proposed a simple heuristic algorithm to find out the suitable configuration for such cases. The heuristic algorithm is in Listing 4.

The heuristic evaluation function *h* in the Listing 4 is a guest of direction to the goal and can be adjusted according to the failure situation. In this research, we have setup the following cases to demonstrate how to implement the heuristic evaluation function (see Table V).

TABLE V
ABNORMAL CASES AND THE HEURISTICS

Case	**Situation**	**Heuristic function**
C1	More power than needs	Smaller generator capacity is better
C2	Lack of power	Larger generator capacity is better

The experimentation results of those cases in Table V are in VI, in which the column **State space** shows the state space of the cases using format “number of states / number of arcs”. Moreover, to double check our proposed heuristic function, the

Listing 4
THE HEURISTIC ALGORITHM

```
function search(start, goal, h)
  frontier := {start};
  while (frontier is not empty) do
    s := get the best state from frontier
    if (s is goal) then
        print the path from the start to s
        exit
    else
        for each s1 in neighbours of s
            frontier += <s1, h(s1)>
        end
    end
  end
  print "no␣such␣solution"
```

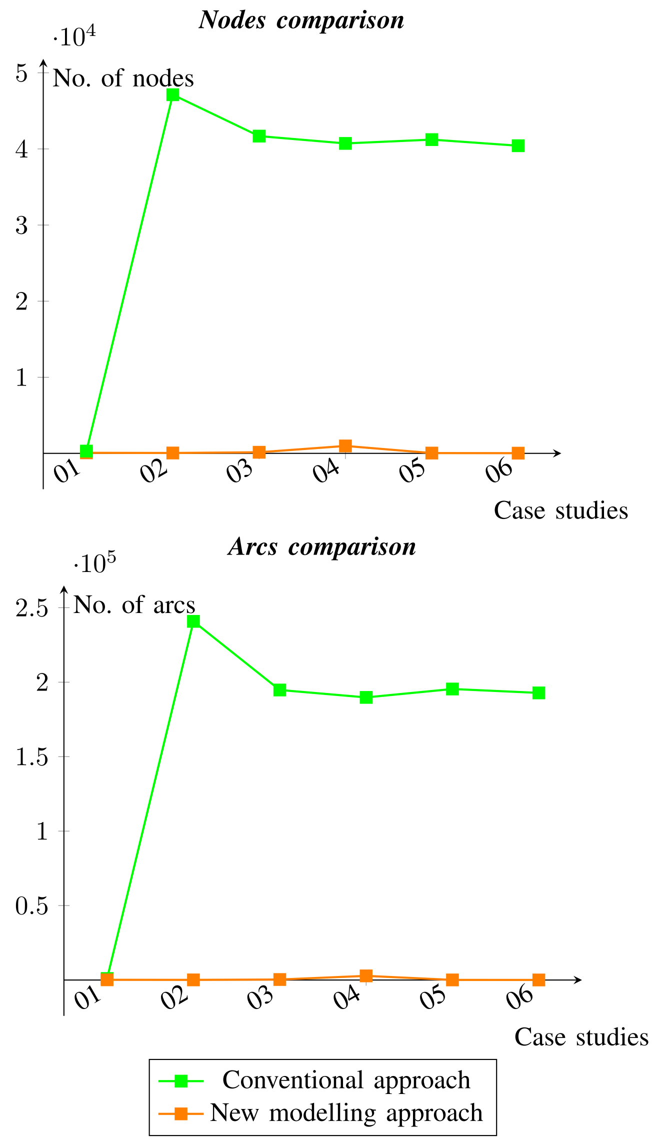

Fig. 10. Comparison between Conventional approach and New modelling approach on State space

reversed of the heuristic functions have also been used for the search. In our opinion, when the heuristic is good, its reversed version should be terrible. Those results are in columns **Heu.** and **Rever.**, respectively.

TABLE VI
EXPERIMENTATION RESULTS

Case	State space	Heu.	Rever.	Configuration
C1	147 / 3639	100ms	1hr	S1:off, S2: on
C2	242 / 9526	30ms	out of time	S1: on, S2: off

A. Discussion

It is easy to see that, by using simple heuristic search, one can find out a solution for each abnormal situations for using in setting up the grids in practice. When all solutions are ready, the setting of the grids for reacting and pro-acting and even detecting/self-healing capacities can be confirmed.

V. Experimentation

A. Experimental setup

In this section, we try to compare the smart grid state spaces generated between the conventional approach and the usage of our proposed approach. Table 3 shows the configuration of the grid in the other four case studies. We increase some of the power of these Generators to show an explosion of state space and running time.

TABLE VII
THE CONFIGURATION OF THE GRID IN FOUR CASE STUDIES.

Case Studies	G1	G2	G3	C1	C2	C3
Case Study 01	2	1	3	3	1	2
Case Study 02	10	5	6	6	9	6
Case Study 03	30	15	18	20	30	13
Case Study 04	50	25	30	35	45	25

B. Experimental results

The experimentation results are represented in Table VIII for the conventional approach and Table IX for our new approach.

For the conventional approach (see Table VIII), the CPN tool can analyze the full state space only when the power capacities of Generators are small as in the first two cases in seasonable amount of time (~ 0 second and 149 seconds). However, when the power capacities of Generators increased, as in case 03 and case 04, the CPN tool can only analyze a part of the state space and took more time (over 900 seconds).

For our new modelling approach (see Table IX), the CPN tool can analyze the full state space for all four cases very fast (almost zero seconds). The generated state spaces are also much more smaller compare the that of the conventional approach.

TABLE VIII
THE RESULTS OF FOUR CASE STUDIES IN CONVENTIONAL APPROACH.

Case	Nodes	Arcs	Run time (sec.)	Status
Case 01	300	1000	~ 0	Full state space
Case 02	47124	240828	149	Full state space
Case 03	42292	198150	834	Partial state space
Case 04	41610	195116	over 900	Partial state space

TABLE IX
THE RESULTS OF FOUR CASE STUDIES IN NEW MODELLING APPROACH.

Case	Nodes	Arcs	Run time (sec.)	Status
Case 01	71	150	~ 0	Full state space
Case 02	48	96	~ 0	Full state space
Case 03	53	106	~ 0	Full state space
Case 04	51	101	~ 0	Full state space

VI. Conclusions

Research on modelling smart grids and verifying their capacities has been raised in decades as it allows engineers to examine and analyse the smart grids from early stage to operation stage. In this paper, a new proposed approach for modelling smart grids using CPN has been presented. It allows engineers to model smart grid topology and elements such as smart generators, consumers, batteries, etc. The advantages of the new approach is that (1) it allow The new approach enables the re-configuration for adapting to the change in the grid layout, and also allows engineers to verify the reacting and pro-acting and even detecting/self-healing capacities of the nets. It has shown that the model is always simple regardless of the complexity of the actual smart grids.

In comparing to the conventional modelling approach, nets constructed from the new approach have smaller state spaces, even thought the grids are complex. However, engineers have to choose between using conventional approach for detail analysis and suffering the complex of the state space and time consuming, and using our approach for overall analysis for fast.

Also, in this text, we have shown that, by searching through the state space, from the initial state to a final state, one can figure out how to configure the smart grid for a situation. Such configurations can be used to demonstrate "smart" capacities of smart grids. When the size and the complexity of the (grid) state space is growing up, the effectiveness of the search has to be studies carefully. This study will be carried out in our near future work.

Last but not least, the tool for engineers to describe the smart grid topology and convert into our CPN models has also been provided in the near future.

Acknowledgment

This research is funded by grant number C2020-20-32 from Vietnam National University - Ho Chi Minh City.

References

[1] S. Patil, G. Zhabelova, V. Vyatkin, and B. McMillin, "Towards formal verification of smart grid distributed intelligence : Freedm case," in *Annual Conference of the IEEE Industrial Electronics Society, IECON 2015 : Yokohama, Japan, 9-12 Nov. 2015*, ser. I E E E Industrial Electronics Society. Annual Conference. Proceedings, 2016, pp. 3974–3979, validerad; 2016; Nivå 1; 2016-11-25 (andbra).

[2] D. Drozdov, S. Patil, C.-W. Yang, G. Zhabelova, and V. Vyatkin, "Formal verification of protection functions for power distribution networks," in *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, 2018, pp. 3550–3555.

[3] S. A. Naseem, R. Uddin, O. Hasan, and D. Fawzy, "Probabilistic formal verification of communication network-based fault detection, isolation and service restoration system in smart grid," *Journal of Applied Logics — IFCoLog Journal of Logics and their Applications*, vol. 5, 01 2018.

[4] K. Le, T. Bui, T. Quan, L. Petrucci, and É. André, "Component-based abstraction of petri net models: An application for congestion verification of wireless sensor networks," in *Proceedings of the Sixth International Symposium on Information and Communication Technology (SoICT 2015)*, December 2015, pp. 342–349.

[5] K. Le, T. Bui, and T. Quan, "State space reduction on wireless sensor network verification using component-based petri net approach," *REV Journal on Electronics and Communications*, vol. 5, no. 3-4, 2016.

[6] K. Le, T. Bui, T. Quan, L. Petrucci, and E. Andre, "Congestion verification on abstracted wireless sensor networks with the wsn-pn tool," *Journal of Advances in Computer Networks*, vol. 4, no. 1, 2016.

[7] K. Le, T. Bui, T. Quan, and L. Petrucci, "A framework for fast congestion detection in wireless sensor networks using clustering and petri-net-based verification," in *Proceedings of the International Workshop on Petri Nets and Software Engineering 2016, Toruń, Poland, June 20-21, 2016*, ser. CEUR Workshop Proceedings, L. Cabac, L. M. Kristensen, and H. Rölke, Eds., vol. 1591. CEUR-WS.org, 2016, pp. 329–334. [Online]. Available: http://ceur-ws.org/Vol-1591/paper21.pdf

[8] K. Le, T. Cao, P. Le, B. Pham, T. Bui, and T. Quan, "Probabilistic congestion of wireless sensor networks: a coloured petri net based approach," *Communications on Applied Electronics*, vol. 7, no. 2, pp. 1–7, May 2017. [Online]. Available: http://www.caeaccess.org/archives/volume7/number2/735-2017652602

[9] T. Bui and A. Nymeyer, "Formal verification based on guided random walks," in *Proc. the 7^{th} Int. Conf. on Integrated Formal Methods (iFM'09)*, ser. LNCS, vol. 5423. Springer-Verlag, Feb 2009, pp. 72–87.

[10] P. T. H. Nguyen and T. H. Bui, "A multiple refinement approach in abstraction model checking," in *Computer Information Systems and Industrial Management, CISIM 2015*, ser. LNCS, K. Saeed and V. Snášel, Eds., vol. 8838. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 433–444.

[11] T. Bui, "Parallelizing random-walk based model checking," *Science and Technology Development Journal*, vol. 18, pp. 108–118, 08 2015.

[12] H. Khai, T. Quan, and T. Bui, "A bitwise-based indexing and heuristic-driven on-the-fly approach for web service composition and verification," *Vietnam Journal of Computer Science*, vol. 4, 09 2016.

[13] K. T. Huynh, T. H. Bui, and Q. T. Tho, "One size does not fit all: Logic-based clustering for on-the-fly web service composition and verification," *International Journal of Web and Grid Services*, vol. 14, no. 3, p. 237–272, jan 2018. [Online]. Available: https://doi.org/10.1504/IJWGS.2018.092579

[14] K. Jensen, "A brief introduction to coloured petri nets," in *Tools and Algorithms for the Construction and Analysis of Systems*, E. Brinksma, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 203–208.

[15] M. Morgan, J. Apt, L. Lave, M. Ilic, M. Sirbu, and J. Peha, "The many meanings of "smart grid"," *SSRN Electronic Journal*, 01 2009.

[16] M. Dotoli, M. P. Fanti, A. Mangini, and W. Ukovich, "Fault detection of discrete event systems using petri nets and integer linear programming," *Automatica*, vol. 45, pp. 2665–2672, 11 2009.

[17] A. Ashouri, A. Jalilvand, R. Noroozian, and A. Bagheri, "Modeling and evaluation of the power system protection using petri nets," in *2010 IEEE International Conference on Power and Energy*, 11 2010, pp. 306–311.

[18] A. Dey, N. Chaki, and S. Sanyal, "Modeling smart grid using generalized stochastic petri net," *Journal of Convergence Information Technology*, vol. 6, 08 2011.

[19] T. M. Chen, J. C. Sanchez-Aarnoutse, and J. Buford, "Petri net modeling of cyber-physical attacks on smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 741–749, 2011.

[20] A. I. Pózna, A. Fodor, M. Gerzson, and K. Hangos, "Colored petri net model of electrical networks for diagnostic purposes," *IFAC-PapersOnLine*, vol. 51, pp. 260–265, 01 2018.

[21] K. Mahmud, A. K. Sahoo, E. Fernandez, P. Sanjeevikumar, and J. B. Holm-Nielsen, "Computational tools for modeling and analysis of power generation and transmission systems of the smart grid," *IEEE Systems Journal*, vol. 14, no. 3, pp. 3641–3652, 2020.

[22] R. Mahmud, R. Vallakati, A. Mukherjee, P. Ranganathan, and A. Nejadpak, "A survey on smart grid metering infrastructures: Threats and solutions," in *2015 IEEE International Conference on Electro/Information Technology (EIT)*, 2015, pp. 386–391.