Project no.:	**IST-FP6-STREP - 027513**
Project full title:	**Critical Utility InfrastructurAL Resilience**
Project Acronym:	**CRUTIAL**
Start date of the project:	**01/01/2006** **Duration: 36 months**
Deliverable no.:	**D8**
Title of the deliverable:	**Preliminary modelling framework**

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)

Contractual Date of Delivery to the CEC:	31/12/2007
Actual Date of Delivery to the CEC:	18/01/2008
Organisation name of lead contractor for this deliverable	LAAS-CNRS
Author(s): M. Kaâniche[4] (Editor), M. Beccuti[6], C. Brasca[1], S. Chiaradonna[3], S. Donatelli[6], F. Di Giandomenico[3], G. Franceschinis[6], K. Kanoun[4], J.-C. Laprie[4], P. Lollini[3], F. Romani[3]	
Participant(s): [4]LAAS-CNRS, [1]CESI-R, [3]CNR-ISTI, [6]CNIT	
Work package contributing to the deliverable:	WP2
Nature:	R
Dissemination level:	PU
Version:	4.0
Total number of pages:	77

Abstract:

This deliverable presents the preliminary version of the CRUTIAL modelling framework aimed at describing interdependencies related failures and evaluating their impact on the dependability and security of information and controlled electricity infrastructures, accounting for both accidental and malicious faults. Two main complementary objectives are followed: 1) the development of qualitative models describing cascading, escalating and common cause failures, where the infrastructures are modelled globally at a high abstraction level; and 2) the development of detailed hierarchical quantitative evaluation models of the infrastructures taking into account their internal architectures and the behaviour of their components resulting from the occurrence of electrical and ICT failures and recoveries.

Keyword list: critical infrastructures, power systems, interdependencies modelling, dependability and security evaluation

DOCUMENT HISTORY

Date	Version	Status	Comments
15/10/2007		Draft	First version of Table of contents
10/12/2007		Draft	First inputs received from partners
18/12/2007	V1.0	Draft	First Complete Draft
21/12/2007	V2.0	Draft	updated Draft
15/01/2008	V3.0	Draft	Final integration and revision
18/01/2008	V4.0	Final	Integration of final comments and production of final version delivered to EC

Table of Contents

1 EXECUTIVE SUMMARY

The CRUTIAL project focuses on two interdependent critical infrastructures: the electrical power infrastructure and the ICT infrastructures supporting management, business, control and maintenance functionality. The project aims to: 1) set up conceptual analysis and assessment models that can be used to identify and analyse interdependencies related failures and to assess their impact on the availability and security of the service delivered to the users, and 2) develop ICT architectural solutions that are well suited to cope with malicious and accidental threats that might affect the dependability and resilience of the considered infrastructures.

This deliverable has been produced in the context of Workpackage 2 that deals with the modelling of interdependencies. The activities of this Workpackage have been structured into two tasks: T2.1 and T2.2.

The first task, T2.1, started in April 2006 and completed at the end of December 2006, was dedicated to the identification and the selection of methodologies that are well suited to address the challenges raised by the resilience modelling, analysis and assessment of the interdependencies between the information infrastructures and the control power infrastructure. The results of this task are reported in deliverable D3 [Kaâniche *et al.* 2007]. Besides discussing the state of the art, this deliverable presented preliminary directions of the modelling framework investigated by CRUTIAL.

The second task, T2.2, started in January 2007 and is planned to last until December 2008. The goal of this task is to develop the main concepts and building blocks of the CRUTIAL modelling framework to support the resilience modelling, analysis and assessment of interdependent information and controlled power infrastructures, accounting for both accidental as well as malicious faults.

This deliverable summarizes the progress achieved in the context of Task 2.2 towards the definition of the CRUTIAL modelling framework based on the preliminary directions defined in Deliverable D3. The CRUTIAL modelling framework follows two main complementary objectives:

1. The development of qualitative models describing the typical failures that are characteristic of interdependent infrastructures, i.e., cascading, escalating and common cause failures. Here the infrastructures are modelled globally without explicitly describing their components behaviours

2. The development of detailed models of the infrastructures taking into account their internal architectures and the behaviour of their components resulting from the occurrence of electrical and ICT failures and recoveries.

The proposed framework is based on a hierarchical modelling approach that accommodates the composition of different types of models and formalisms, and generic building blocks, modelling the specific types of interdependencies, interactions or failure behaviours that are characteristic of interdependent information and controlled power infrastructures.

This deliverable is structured into four sections.

Section 2 gives an overview of the main contributions proposed in the context of the CRUTIAL modelling framework for describing, modelling and assessing interdependencies related failures in the context of electrical and ICT infrastructures.

Section 3 deals with the qualitative modelling of interdependencies related failures. It is structured into three main subsections. Section 3.1 presents unified models of cascading, escalating and common cause failures that are well suited to describe the impact of accidental threats as well as of malicious threats. A formalisation of the interdependencies related failures described in the unified model is presented in Section 3.2. Finally, a

compositional modelling framework based on stochastic Petri nets that support the qualitative modelling of interdependencies related failures is presented in Section 3.3.

Section 4 deals with the detailed modelling of the electrical and ICT infrastructures taking into account interdependencies related failures, in order to obtain quantitative measures characterizing their resiliency. A hierarchical modelling framework based on stochastic activity networks is presented. Basic building blocks of elementary entities of the investigated infrastructures and their dependencies, implemented with the Möbius tool are described. The objective is to develop generic reusable models, accounting as much as possible for internal dynamics of the represented entities as well as dependencies among them, and for generic fault and propagation conditions, in accordance with effective metrics defined to quantitatively assess the impact of interdependencies.

Finally, Section 5 presents the main conclusions and outlines future work.

2 OVERVIEW OF THE CRUTIAL MODELLING FRAMEWORK

Critical infrastructures are complex collections of interacting systems and components communicating through multiple heterogeneous networks. The interactions between these components and systems need to be analyzed carefully to understand and characterize the interdependencies. An *interdependency* is a bidirectional relationship between two infrastructures through which the state of each infrastructure influences or is correlated to the state of the other. More generally, two infrastructures are interdependent if each is dependent on the other.

Interdependencies increase the vulnerability of the corresponding infrastructures as they give rise to multiple error propagation channels from one infrastructure to another that make them more prone to exposure to accidental as well as to malicious threats. Consequently the impact of infrastructure components failures and their severity can be exacerbated and are generally much higher and more difficult to foresee, compared to failures confined to single infrastructures. As an example, most major power grid blackouts that have occurred in the past were initiated by a single event (or multiple related events such as a power grid equipment failure that is not properly handled by the SCADA, i.e., Supervisory Control And Data Acquisition, system) that gradually leads to cascading failures and eventual collapse of the entire system [Pourbeik *et al.* 2006].

There is a wide consensus that developing modelling frameworks for understanding interdependencies among critical infrastructures and analysing their impact is a necessary step for building interconnected infrastructures on which a justified level of confidence can be placed with respect to their robustness to potential vulnerabilities and disruptions. Modelling can provide useful insights into of how components failures might propagate and lead to cascading, or escalating failures in interdependent infrastructures, and assess the impact of these failures on the service delivered to the users.

In the context of CRUTIAL, we focus on two interdependent infrastructures: the electric power infrastructure and the information infrastructures supporting management, business, control and maintenance functionality. We concentrate on cascading, escalating and common-cause failures, which correspond to the main causes of interdependency-related failures [Rinaldi *et al.* 2001]:

1. *A* cascading failure occurs when a disruption in one infrastructure causes the failure of a component in a second infrastructure, which subsequently causes a disruption in the second infrastructure.
2. An *escalating failure* occurs when an existing disruption in one infrastructure exacerbates an independent disruption of a second infrastructure, generally in the form of increasing the severity or the time for recovery or restoration of the second failure.

3. A *common cause failure* occurs when two or more infrastructure networks are disrupted at the same time: components within each network fail because of some common cause.

As discussed in deliverable D3 [Kaâniche *et al.* 2007], there has been extensive work on the modelling of individual infrastructures and various methods and tools have been developed to predict the consequences of potential disruptions within an individual infrastructure. However, the modelling and evaluation of interdependent infrastructures is still at an exploratory stage. The modelling activities carried out in CRUTIAL aim at contributing at filling this gap taking into account in particular: a) the three types of failures that are characteristic of interdependent infrastructures (cascading, escalating, and common-cause failures), b) various classes of faults that can occur, including accidental as well as malicious threats, c) the temporal and structural characteristics of the power and information infrastructures investigated.

A major challenge in the project lies in the complexity of the modelled infrastructures in terms of largeness, multiplicity of interactions and types of interdependencies involved. To address this problem, a number of abstractions and appropriate approaches for composition of models are necessary. The aim is therefore to produce, from conceptual analyses, generic models that can be refined, instantiated and composed according to hierarchical modelling approaches. Resorting to a hierarchical approach brings benefits under several aspects, among which: i) facilitating the construction of models; ii) speeding up their solution; iii) favouring scalability; iv) mastering complexity by handling smaller models through hiding at one hierarchical level some modelling details of the lower one. Important issues are how to abstract all the relevant information of one level to the upper one and how to compose the derived abstract models.

Following these principals, two complementary types of models are explored in CRUTIAL to fulfil the objectives stated above:

- *High-level qualitative models* describing failure scenarios resulting from mutual interdependencies between the electrical and the information infrastructures, where the infrastructures are modelled globally without explicitly modelling their components.

- *Detailed hierarchical quantitative evaluation* models based on stochastic processes, taking into account explicitly the structure of the electrical and the information infrastructures and the behaviour of their components resulting from the occurrence of failures, their propagation and the application of recovery scenarios.

In the following we summarize the progress and the main contributions obtained so far considering these two types of models. The detailed descriptions of these contributions are presented in Section 3 and in Section 4, respectively.

2.1.1 Contributions related to the high-level qualitative models

The high-level qualitative models are very relevant to characterize interdependencies-related failures and to understand how cascading, escalating and common cause failures might occur and potentially lead to blackouts. We have presented in deliverable D3 preliminary qualitative models that take into account accidental threats. Analogous models have been also proposed for malicious threats. In this deliverable, we introduce three significant contributions and extensions of the qualitative models presented in deliverable D3.

The first contribution concerns the definition of a *unified model* that generalizes the two separate models proposed in D3, taking into account accidental and malicious threats in an integrated way. This model is presented in Section 3.1.

The second contribution detailed in Section 3.2 concerns the development of a formal setting based on a new class of automata called Dependent Automata, in which the three types of interdependencies related failures and the unified model presented in Section 3.1 can be

rigorously and precisely defined. The proposed formalism is well suited to model dependencies and chains of cause-effects that can potentially exist between states, events, or a mix of the two. Also, it is shown that the whole set of dependencies described in the unified model can be represented through two basic rules: synchronisation and effect function.

The development of compositional modelling frameworks is also important to allow the separation of concerns and the reusability of the proposed models in other contexts. The third contribution related to the high-level qualitative models of interdependencies, detailed in Section 3.3, aims to fulfil this objective. It concerns the elaboration of a compositional representation of the qualitative models based on Petri nets. So far, the results obtained are based on the preliminary models presented in deliverable D3. Extension of the approach to generate the unified models presented in Section 3.1 is under investigation.

2.1.2 Contributions related to the hierarchical quantitative evaluation modelling

In Deliverable D3, we have set up the preliminary concepts by identifying the prominent aspects of the hierarchical modelling framework aimed at the detailed description of the electrical infrastructure and the information infrastructure components and their failure behaviours with the purpose of evaluating quantitative measures characterizing the impact of interdependencies-related failures on the resilience of the delivered service. This deliverable summarizes the new developments carried out during the last year and describes the first implementation of the proposed framework using the Möbius tool. The proposed building blocks and modelling templates are designed to be generic and reusable, accounting as much as possible for the internal dynamics of and dependencies among the modelled entities. An approach to compose the submodels and building blocks in a hierarchical way is also proposed. Also, a simulator based on stochastic models has been developed in order to capture some failure characteristics related to the dynamics of the electrical infrastructure components.

3 QUALITATIVE MODELS OF INTERDEPENDENCIES

3.1 Unified models and application to scenarios

A preliminary framework for modelling interdependencies between the information infrastructure and the electricity infrastructure, taking into account failures in both infrastructures, has been defined in deliverable D3. In this preliminary framework, for the information infrastructure, the model related to accidental faults on one hand, and the model related to malicious attacks on the other hand, have been built separately, even though following similar approaches. However, the information infrastructure is prone to both accidental faults and malicious attacks, and a unified modelling framework is necessary to reflect the behaviour of the information infrastructure and the electricity infrastructure and their interdependencies under both classes of faults.

The aim of this section is to present a unified modelling approach of the interdependencies between the information infrastructure and the electricity infrastructure, taking into account accidental faults and malicious attacks. In addition, to make this approach tractable, we use a unified terminology for events occurring in both infrastructures and states.

Section 3.1.1 presents the updated modelling approach. It focuses, as in the preliminary framework, on two interdependent infrastructures: the electric power infrastructure and the information infrastructures supporting management, control and maintenance functionality. It addresses modelling and analysis of interdependency-related failures between these

infrastructures. Section 3.1.2 shows the link between the model presented in Section 3.1.1 and the scenarios presented in deliverable D1.

3.1.1 Unified modelling approach

As in deliverable D3, we concentrate on cascading, escalating and common-cause failures, because they correspond to the main classes of interdependency-related failures. We model the infrastructures globally, not explicitly modelling their components. The models presented are qualitative ones. They describe scenarios that are likely to take place when failures occur. The models are built based on assumptions related to the behaviour of the infrastructures as resulting from their mutual interdependencies.

In this section, we first present the events that may occur in each infrastructure, and the resulting states. Then we address successively i) cascading failures, ii) cascading and escalading failures, and iii) common mode failures.

3.1.1.1 Events and states of the two infrastructures

For the sake of clarity, and in order to avoid any confusion between the events affecting the two infrastructures, we use specialized but similar terms for the two infrastructures events as indicated by Table 1.

Table 1: Events and states of each infrastructure when considered alone

	Information Infrastructure	Electricity Infrastructure
Events	i-failure	e-failure
	i-restoration	e-restoration
States	i-working	e-working
	partial i-outage	partial e-outage
	i-weakened	e-weakened
		e-lost

The weakened state of one infrastructure corresponds to a state in which this infrastructure performs its functions in a degraded mode of operation due a failure in the other infrastructure.

The high interconnectivity of the electricity network, and the variety of functions and components of the information infrastructure make unlikely total outage of the infrastructures.

The e-lost state corresponds to the propagation of e-failures within the electricity infrastructure leading to loosing its control.

Assumption concerning i-failures and i-states

- i-failures, can be **signalled** (i.e., detected) or **unsignalled**.
- After signalled i-failures, the information infrastructure is in **a partial i-outage state**: the variety of functions and components of the information infrastructure, and its essential character of large network make unlikely total outage.
- Unsignalled failures bring the information infrastructure into a **latent error state** in which parts of the information infrastructure have a non detected i-failure. Two classes of latent error states can be distinguished:

- **Passive latent error state**: the unsignalled i-failure prevents the information infrastructure from monitoring the state changes in the electricity infrastructure (i. e., without detection of state changes and, as a consequence without any reaction to occurring e-failures); an e-failure may remain unnoticed.

- **Active latent error state:** the unsignalled i-failure prevents the information infrastructure from perceiving correctly the state of the electricity infrastructure; two situations are to be considered:

  i) **optimistic latent error state** in which the information infrastructure leads to unnecessary, and unnoticed configuration changes in the electricity infrastructure, and still declares it as working (i.e., the electricity infrastructure appears as working when weakened),

  ii) **pessimistic latent error state** in which the information infrastructure declares the electricity infrastructure as partially in outage (i.e., the electricity infrastructure appears as partially in outage) when it is actually working. As a consequence, some configuration changes are performed in the electricity infrastructure that may lead it to an e-weakened state. The difference between this state and the previous one is that in this state there is awareness of the actions performed on the electricity infrastructure, while in the previous state there is no awareness of the actions performed in the electricity infrastructure by the information infrastructure.

- Signalled i-failures may take place when the information infrastructure is in latent error states leading it to a partial i-outage state.

- When the information infrastructure is in a partial i-outage state, i-restoration is necessary to bring it back to an i-working state

The four classes of i-failures are summarized in Figure 1. It is worth to note that these failures may result either from accidental faults or malicious attacks. Considering malicious attacks, signalled failures result from what is usually referred to as perceptible attacks (those creating detected damages), while unsignalled failures result from what is usually referred to as deceptive attacks (those provoking unperceived malfunctions).

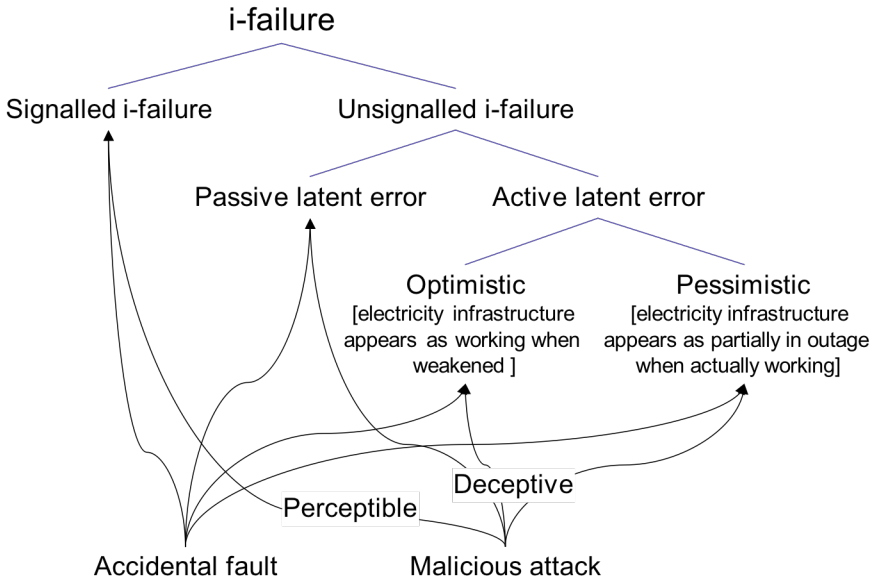


Figure 1: i-failure classification

Figure 2-a gives the state machine model of the information infrastructure taking into account its own failures. It is noteworthy that all states are presented by several boxes, meaning that a state corresponds in reality to a group of different states that are considered as equivalent with respect to the classification given in Table 1. For example all states with only one busbar isolated in the electricity infrastructure, due to a pessimistic latent error in the information infrastructure, can be considered as equivalent irrespective of which busbar is isolated.

It is worth to mention that signalled i-failures that are i) without any impact on the electricity infrastructure and that are ii) processed automatically, are not represented in the figure. As we are mainly concerned by modelling interdependencies, and due to the very short processing time, we assume that the system remains in an i-working state.

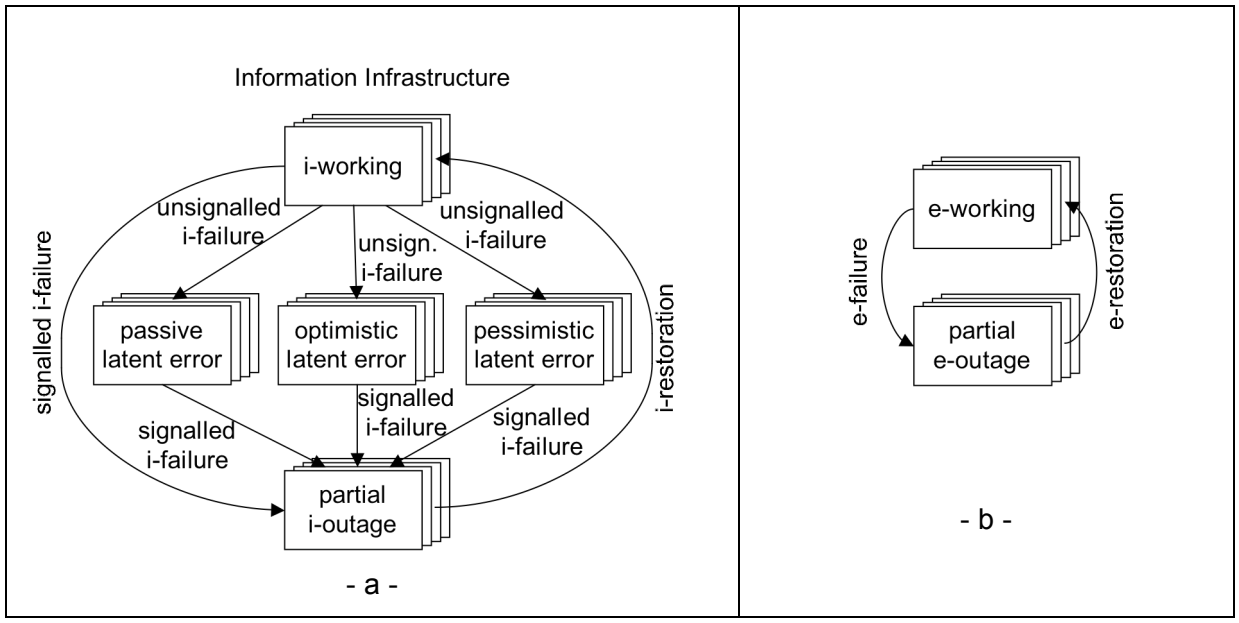


Figure 2: Infrastructure states as resulting from their own failures

Assumption concerning e-failures and e-states

- e-failures bring the electricity infrastructure into **a partial e-outage state**; we assume that the role of the information infrastructure, if it is in an i- working state, is to avoid a total e-outage (or total loss) of the electricity infrastructure, thanks to the high interconnectivity of the electricity network.

- When the electricity infrastructure is in a partial e-outage state, an e-restoration is necessary to bring it back to an e-working state.

- The e-working state gathers in reality several working configurations of the electricity infrastructure; these configurations correspond to operational configurations as resulting from the operational demand of the electricity infrastructure, without any constraints imposed by the information infrastructure leading to the failure of the latter. In such working states the electricity infrastructure may change its configuration regularly, without any failure in both infrastructures. For example, configuration changes take place very often for scheduled (or preventive) maintenance purposes, for which some parts of the electricity infrastructure have to be isolated for inspection and repair if required.

The state machine model of the electricity infrastructure is given in Figure 2-b.

3.1.1.2 Cascading failures

A cascading failure occurs when a failure in one infrastructure prevents the other infrastructure from implementing its functions nominally. We will first analyse the impact of the electricity infrastructure failures on the states of the information infrastructure, then the impact of the information infrastructure failure on the states of the electricity infrastructure.

Impact of the electricity infrastructure failures on the states of the information infrastructure

Even though some e-failures do not have any impact on the states of the information infrastructure, some other e-failures lead the information infrastructure to a state in which parts of the information infrastructure can no longer implement their functions, although they are not failed, due to constraints originating from the failure of the electricity infrastructure, e.g., shortage of electricity supply of unprotected parts (e.g., the business part of the infrastructure). Such a state is referred to as an **i-weakened state.** Figure 3-a shows the constraint that the electricity infrastructure puts on the information infrastructure assuming that the latter is in an i-working state.

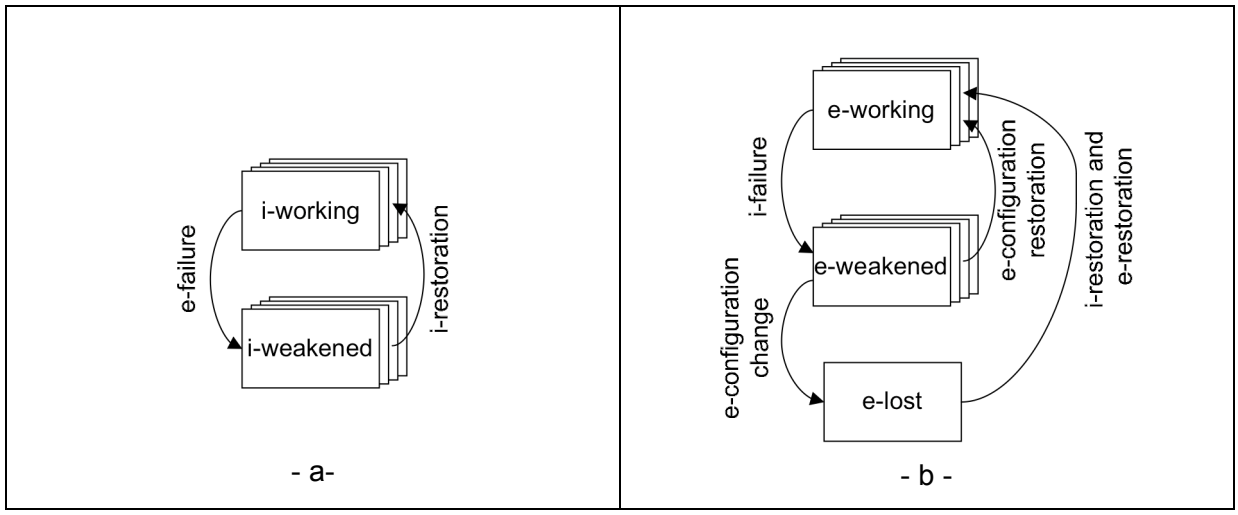


Figure 3: Constrained states of the infrastructures due to interdependencies

Impact of the information infrastructure failure on the states of the electricity infrastructure

Passive latent errors in the information infrastructure do not have any impact on the electricity infrastructure.

When the information infrastructure is in an optimistic error state, it puts some constraints on the electricity infrastructure: it performs unduly isolations, or unnecessary off-line trips of production plants or of transmission lines. The capability of the electricity infrastructure is degraded: lower performance, configuration changes, possible manual control, etc.. The associated state is referred to as **e-weakened state.** From an e-weakened state, an e-configuration restoration leads the electricity infrastructure back into an e-working state (e-restoration is not required because no e-failures occurred in the electricity infrastructure). Accumulation of untimely configuration changes, may lead to **e-lost state** (i.e., a blackout state), from which an e-restoration is required to bring back the electricity infrastructure into an e-working state.

In a pessimistic latent error state, the information infrastructure perceives the electricity infrastructure as being in a partial e-outage while it is working. As a consequence some configuration changes in the electricity infrastructure are performed, leading it to an e-weakened state. Accumulation of configuration changes may lead the electricity infrastructure into e-lost state.

Hence, both active latent error states (optimistic and pessimistic) may lead to cascading failures.

Figure 3-b shows the constraint that the information infrastructure puts on the electricity infrastructure when the latter is in an e-working state. i-failures correspond to signalled or unsignalled i-failures leading to optimistic latent error state.

Table 2 and Table 3 summarize and complement the states and events of each infrastructure, taking into account cascading events, as described above.

Table 2. States and events of the information infrastructure

Events	
Signalled i-failure	Detected i-failure
Unsignalled i-failure	Undetected i-failure
i-restoration	Action for bringing back the information infrastructure in its normal functioning state after i-failure(s) occurred
States	
i-working	The information infrastructure ensures normal control of the electricity infrastructure
Passive latent error	Parts of the information infrastructure have an i-failure, which prevents the information infrastructure from detecting the state changes in the electricity infrastructure; an e-failure may remain unnoticed
Active latent error	Parts of the information infrastructure have an i-failure, that prevents the information infrastructure from perceiving correctly the state of the electricity infrastructure: - optimistic latent error: the electrical infrastructure appears as working while it is weakened - pessimistic latent error: the electrical infrastructure appears as in partial e-outage while it is working or weakened
Partial i-outage	Parts of the information infrastructure have knowingly an i-failure.
i-weakened	Parts of the information infrastructure can no longer implement their functions, due to constraints originating from e-failures,

Table 3. States and events of the electricity infrastructure

Events	
e-failure	Malfunctioning of elements of the power grid: production plants, transformers, transmission lines, breakers, etc.
e-restoration	Actions for bringing back the electricity infrastructure in its normal functioning state after e-failure(s) occurred. Typically, e-restoration is a sequence of configuration change(s), repair(s), configuration restoration(s)
e-configuration change	Change of configuration of the power grid that are not immediate consequences of e-failures, e.g., off-line trips of production plants or of transmission lines
e-configuration restoration	Act of bringing back the electricity infrastructure in its initial configuration, when configuration changes have taken place
States	
e-working	Electricity production, transmission and distribution are ensured in normal conditions
Partial e-outage	Due to e-failure(s), electricity production, transmission and distribution are no longer ensured in normal conditions, they are however ensured in degraded conditions
e-lost	Propagation of e-failures within the electricity infrastructure led to loosing its control, i.e., a blackout occurred.
e-weakened	Electricity production, transmission and distribution are no longer ensured in normal conditions, due to i-failure(s) that constrain the functioning of the electricity infrastructure, although no e-failure occurred.

3.1.1.3 Escalating and cascading failures

An escalating failure occurs when an existing outage in one infrastructure exacerbates an independent outage in the other infrastructure, increasing its severity (escalation in severity) or increases the time for restoration from this outage (escalation in restoration time).

Escalating failures may thus take place only after the occurrence of at least one failure in each infrastructure, or an unsignalled active i-failure (pessimistic or optimistic latent error states) followed by successive e-configuration changes (either by the information infrastructure or by the operator) leading the electricity infrastructure into an e-lost state.

Figure 4 gives the state machine model of the infrastructures, taking into account the constraints of each infrastructure on the other one.

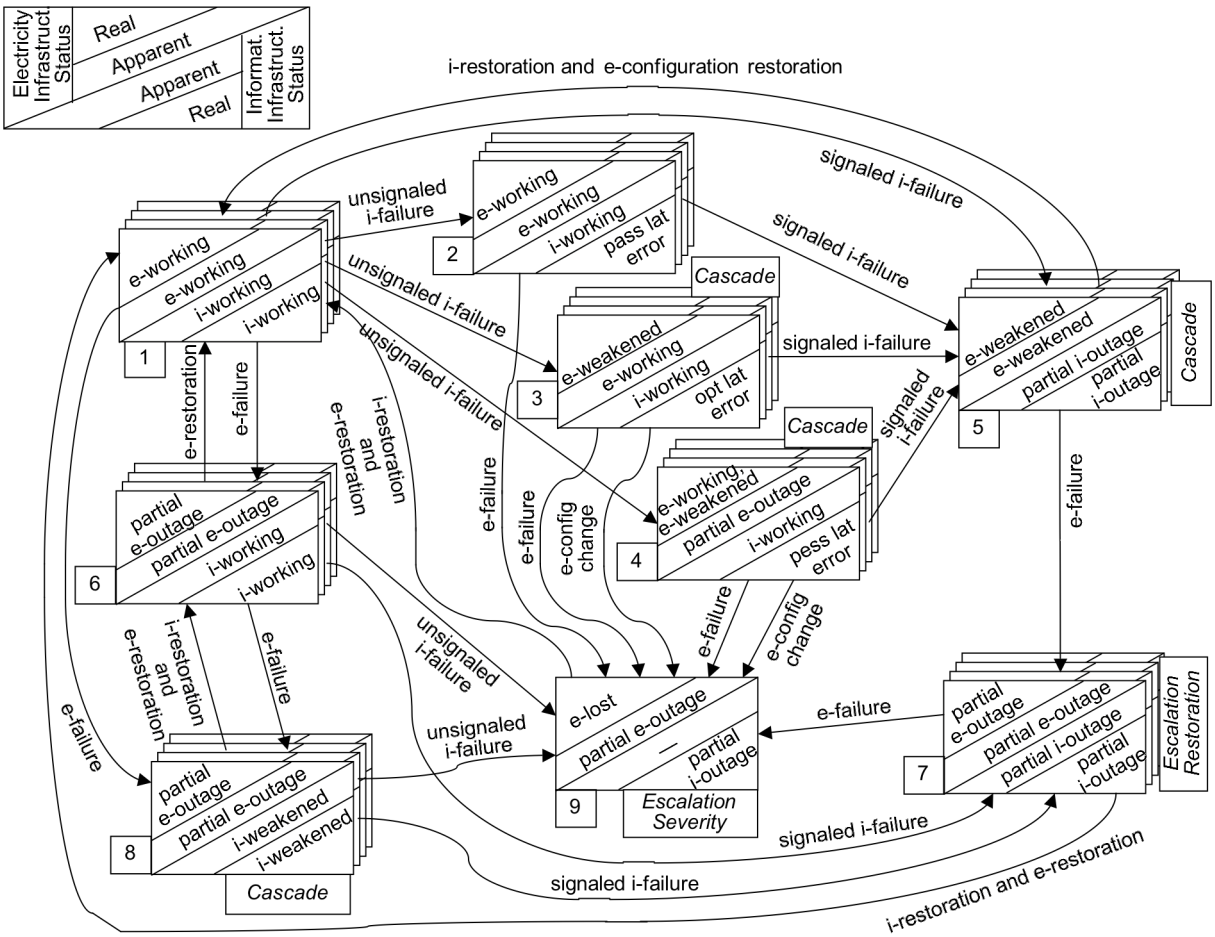


Figure 4: State machine of the infrastructures when considering cascading and escalating outages

The states are described in terms of the statuses of both infrastructures. A distinction has to be performed for both infrastructures between their real status and their apparent status. For the electricity infrastructure, the apparent status is as reported by the information infrastructure.

Both cascading failures (states 3, 4, 5, 8) and escalating ones are evidenced, with a distinction of consequences of the latter in terms of time to restoration (state 7) and of severity (state 9). Dependency of the electricity infrastructure upon the information infrastructure is illustrated by the need for both i- and e-restoration from states 7and 9.

In state 2, the information infrastructure appears as being in an i-working state while it is in a passive latent error state. This state is potentially dangerous as the occurrence of an e-failure brings the infrastructures into state 9 (escalation outage).

In state 3, the effects of the unsignalled active i-failures are: i) the information infrastructure looks like working while it is in an optimistic latent error state due to the unsignalled active i-failure, and ii) it performs some configuration changes in the electricity infrastructure leading it to an e-weakened state without informing the operator about the state of the electricity infrastructure, for whom the electricity infrastructure appears if it were working. Accumulation of configuration changes by the information infrastructure may lead the electricity infrastructure into an e-lost state (state 9).

In state 4, the effects of the unsignalled active i-failure are: i) the information infrastructure looks like working while it is in a partial i-outage state due to the unsignalled active i-failure, and ii) it signals wrongly that the electricity infrastructure is in partial i-outage, and as consequence iii) the operator performs some configuration changes are performed in the electricity infrastructure leading it to an e-weakened state. Accumulation of configuration changes may lead the electricity infrastructure into e-lost state (state 9).

The difference between states 3 and 4 is that in state 4 there is awareness about the actions performed on the electricity infrastructure, while in state 3 there is a lack of awareness of such a situation.

When the infrastructures are in states 2, 3, or 4, after detection of an i-failure, the apparent states of the infrastructures become identical to the real ones (state 5), in which i-restoration and e-configuration restoration are necessary to bring back the infrastructures to their working states. The occurrence of an e-failures in state 5 brings the infrastructures into state 7.

In state 6, an e-outage has occurred and it is being processed by the information infrastructure that is in an i-working state.

State 8 is entered either from state 1 or state 6, following the occurrence of an e-failure that has an impact on the information infrastructure.

Evolution from states 6 and 8 are the same:

- Accumulation of e failures my lead the information infrastructure into an i-weakened state, state 8.

- Occurrence of an unsignalled i-failure brings the two infrastructures into an escalation outage state, state 9, in which the consequences of the e-failure are more severe than those in states 6 and 8.

From state 7 the occurrence of an e-failure brings the infrastructures in state 9.

It is worth to mention that when restoration of the two infrastructures is required, the restoration actions are usually performed in a coordinated way. Restoration of parts of one infrastructure may help restoration of some parts in the other one and vice-versa. This is also true for e-configuration restoration and i-restoration.

A noteworthy example of transitions from states 1 to 2, and from 2 to 9 relates to the August 2003 blackout in the USA and Canada: the failure of the monitoring software was one of the immediate causes of the blackout, as it prevented confining the electrical line incident, before its propagation across the power grid [US-Canada 2004].

3.1.1.4 Common cause failures

Figure 5 gives a model with respect to common-cause failures that would occur when the infrastructures are in normal operation, bringing the infrastructures into states 7 or 9 of Figure 4, i.e., to escalation failures. Should such failures occur in other states of the infrastructures of Figure 4 model, they would also lead to states 7 or 9.

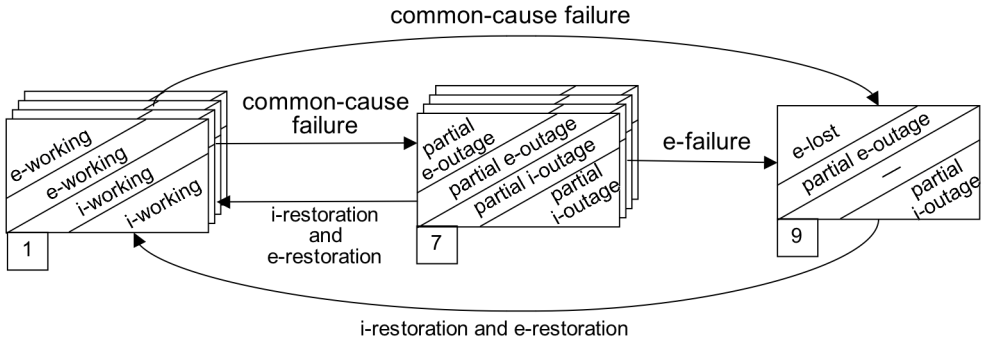


Figure 5: Common-cause failures model

Considering common-cause failures does not introduce additional states, they however add direct transitions from already existing states that do not exist when considering only cascading and escalating failures. The states of the resulting model become almost totally interconnected.

3.1.2 Application to scenarios

In this section we illustrate the unified model using three examples of scenarios among those defined in deliverable D2 [Garrone *et al.* 2007]. The selected examples illustrate different types of failures affecting the information infrastructures (considering malicious and accidental threats) or affecting the electricity infrastructure.

3.1.2.1 Scenario 1: DSO Teleoperation

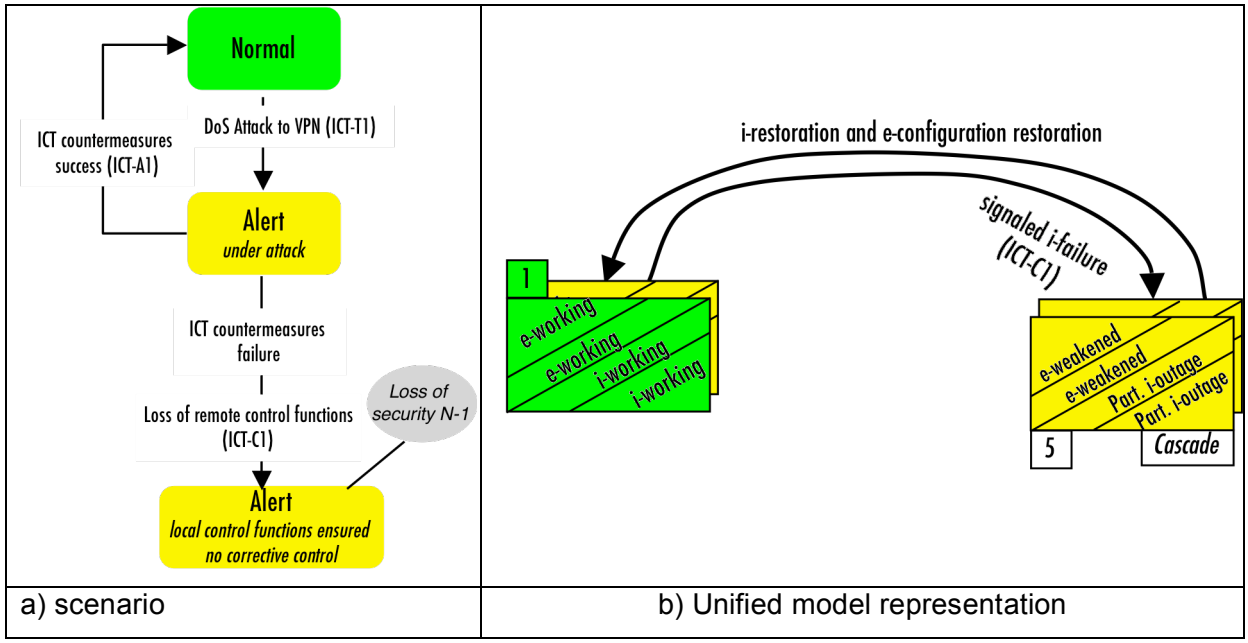
This scenario considers the possible cascading effects of ICT threats to the DSO (Distributed System Operator) communication channel among area control centres and their supervised substations. Two types of malicious threats are considered: Denial of Service (DoS) attacks and Intrusions.

a) DoS attacks

We consider the case of a DoS attack to the VPN (Virtual Private Network) connecting a substation to the ATS (Area Telecontrol System) of the area control centre. This attack reduces the communication bandwidth causing delay or failure in the delivery of status information to the ATS and in the reception of commands from the ATS with consequent partial or complete loss of remote control functions. Local control functions are not affected.

Figure 6-a represents the scenario as described in deliverable D2, and Figure 6b illustrates one possible interpretation of this scenario using the unified qualitative model. It is assumed here that in states Normal and Alert under attack in Figure 6-a, the electrical system and the information system are still able to accomplish their functions despite the fact that a DoS attack is in progress. This is because If the attack succeeds the failure will be finally perceived. This is represented in the unified model of Figure 6-b by the event corresponding to the transition of the information infrastructure from the i-working state to the i-Partial-outage state. Also, it is assumed that the attack affects the state of the electricity

infrastructure leading it the e-weakened state. From state 5, i-restoration and e-configuration restoration actions are needed to recover the initial working state 1.



a) scenario | b) Unified model representation

Figure 6: DoS attack to VPN

b) Intrusion into the Centre/Substation communication network

In this scenario we consider the case of an intrusion targeting the centre/substation communication network followed by the execution of faked commands. As in the previous case, it is assumed here that in states Normal and Alert under attack in Figure 7-a, the electrical system and the information system are not affected as long as the intrusion attempt does not succeed (i.e., they remain in a working state). When the intrusion is successful, two possible alternatives could occur depending on the actions carried out by the attacker and how they are perceived by the information infrastructure. The first alternative corresponds to the occurrence of an unsignalled i-failure leading the information infrastructure to an optimistic latent error state: unduly e-configuration changes are performed that lead the electricity infrastructure to an e-weakened state while it appears as working from the information infrastructure side. The second alternative corresponds to the occurrence of an unsignalled i-failure leading the information infrastructure to a pessimistic latent error state: the electricity infrastructure is in an e-working state while it is perceived by the information infrastructure as in partial outage as a result of the malicious actions performed by the intruder on the ICT.

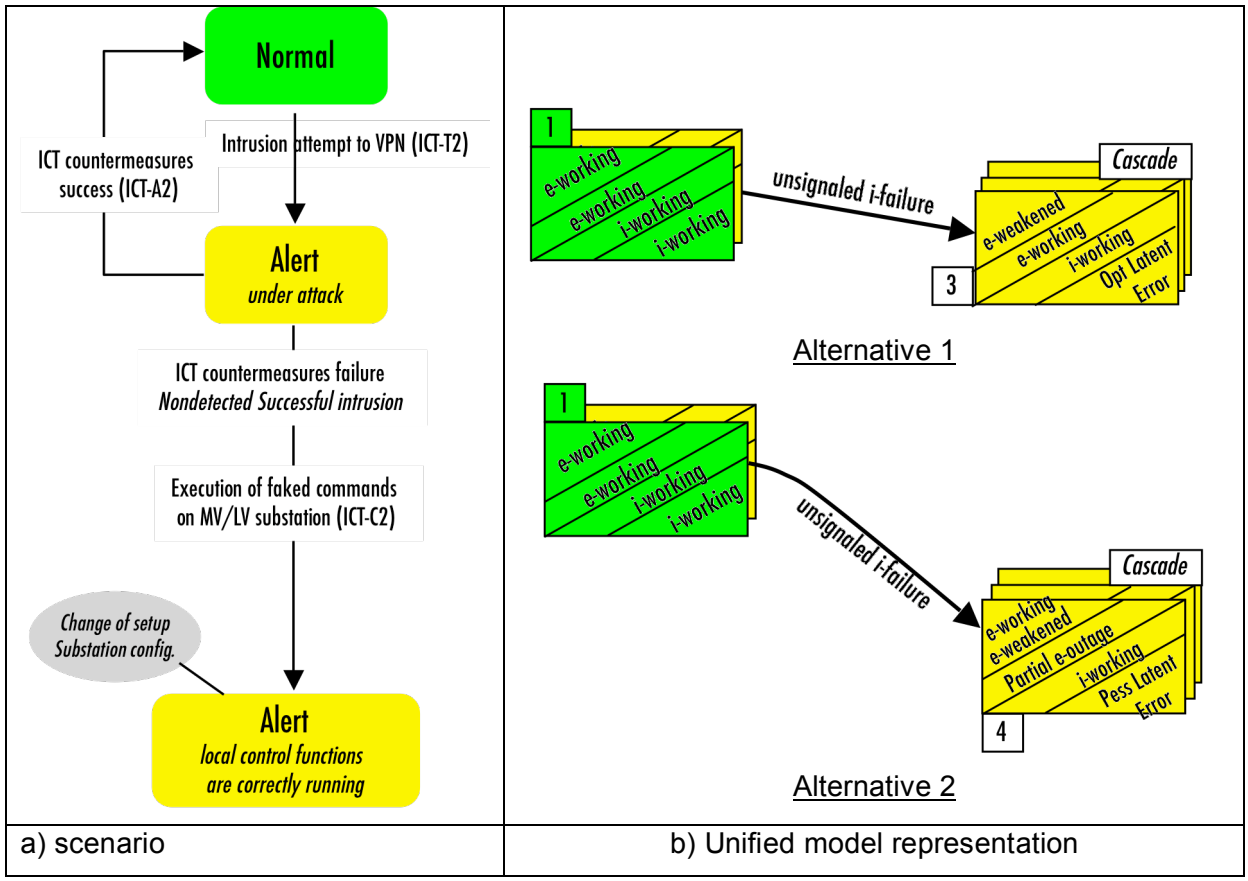


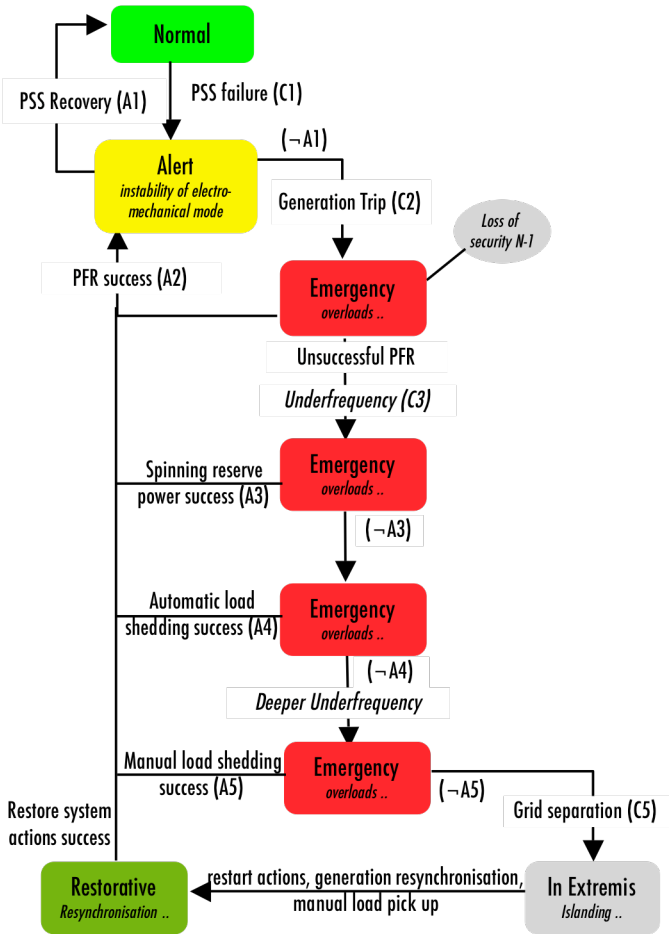
Figure 7: Intrusion into the Centre/Substation communication network

3.1.2.2 Scenario 6-b: instability of electromechanical mode due to PSS-failure

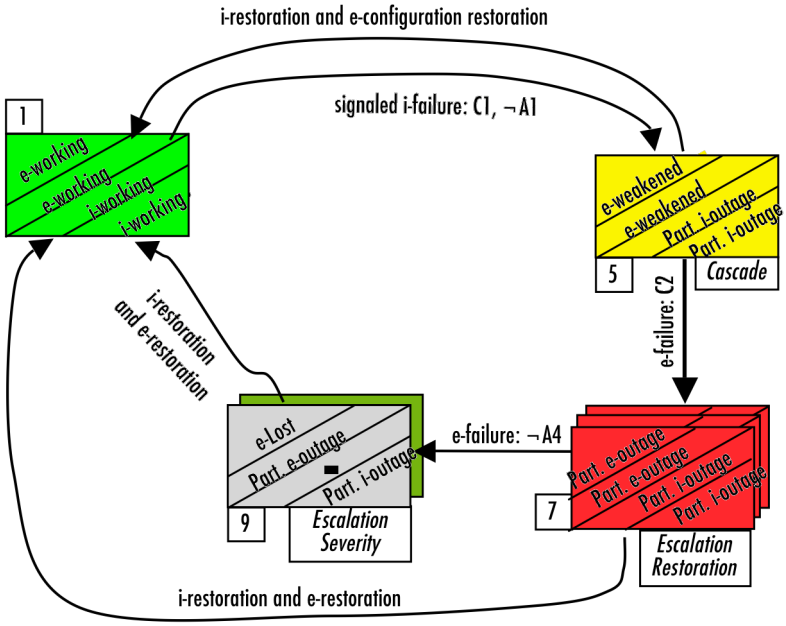
This scenario considers specific abnormal operation conditions of the system composed of a power plant and the transmission grid, which are related to the electromechanical oscillation mode. This mode can become unstable for example as a result of wrong parameters set into the Automatic Voltage Regulator (AVR), in particular the power system stabiliser (PSS), due to an accidental failure or an intrusion through the AVR Human Machine Interface, or through the PQR or the GENCO-SCADA system. As a consequence, erroneous regulation signals could be set leading to the instability of the electromechanical mode.

As illustrated in Figure 8, it is assumed that this scenario is initiated by the occurrence of a signalled i-failure leading the system to an Alert state with consequences on the information infrastructure (Partial i-outage) and the electricity infrastructure (e-weakened state). This corresponds to the transition from State 1 to State 5 in the unified model.

The instability of the electromechanical mode could provoke the trip of the generation group under consideration. This corresponds to the occurrence of an e-failure from state 5 leading the system into an emergency state where both the electrical infrastructure and the information infrastructure are in Partial outage states (State 7). It is noteworthy that in the unified model, we do not model explicitly the scenario corresponding to the successful accomplishment of the i-restoration action from the alert state to the Normal state before the generation trip occurs.



a) scenario



b) unified model

Figure 8: Instability of electromechanical mode due to PSS-failure

The first action to compensate the lost power group (A2) is constituted by the Primary Frequency Regulators (PFR) of the other groups of the power plant. If successful (e-restoration), and the i-failure that initiated this scenario is also recovered the system gets back to the Normal state. This corresponds to the global transition of the unified model from State 7 to State 1. If the PFR actions fail, further degradation of the electricity infrastructure state could be observed due to power instability related to under frequency contingencies (C3). This situation could be recovered through spinning reserve power management actions. If this is still not sufficient, a deeper underfrequency situation will be observed. Then automatic or manual load shedding actions will be needed. If the system still cannot be recovered, the situation becomes very critical and the system reaches its most degraded state where islanding will need to be performed. This corresponds to the occurrence of an e-failure from State 7 to State 9 from which a global e-restoration and i-restoration will be required to recover the Normal state situation.

3.1.2.3 Scenario 8:Transmission grid short circuit with loss of synchronism

This scenario shows interdependencies among different power plants due to the occurrence of a short circuit in the transmission grid (e-failure) that propagates to various Power Plants causing the loss of synchronism of some generators. This situation requires an intervention of the TSO to avoid the occurrence of cascading effects due to the loss of synchronism.

The main ICT function, allowing the electrical power system to recover from alert to emergency due to the short circuit, is a centralised protection/control function. It is based on the acquisition of measurements and signals from the different power plants; it identifies the generators to be tripped in order to avoid the loss of synchronism and sends the appropriate trip commands to the interested power plant controllers. The whole function must be completed within 400-500 milliseconds to avoid loss of synchronism. Possible i-failures may be the following: lack of measurements, lack of signals, loss of automatic generator control, loss of centralised control, loss of regulation functions.

This scenario and the corresponding unified model are described in Figure 9. The occurrence of the e-failure corresponds to the transition from State 1 to State 6.The occurrence of an i-failure leads to the transition from State 6 to State 7. If the system is still unable to recover, it reaches the blackout state represented by state 9. Form there i-restoration and an e-restoration actions will be required to get the system back to the Normal State.

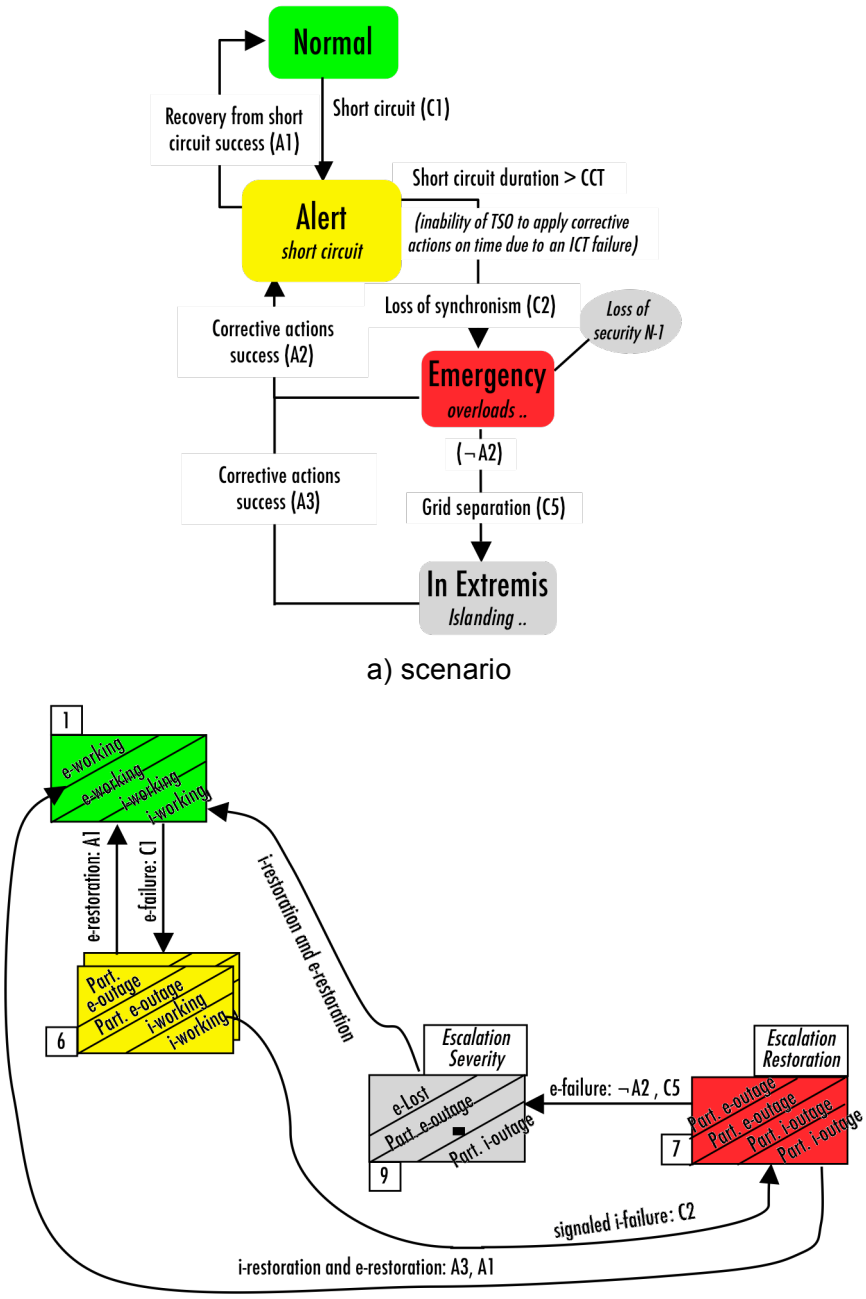


Figure 9: Transmission grid short circuit with loss of synchronism

3.2 FORMALISATION OF INTERDEPENDENCIES

Although there are many works dealing with interdependency [Kaâniche *et al.* 2007], there is not a single definition of what interdependency means in a formal setting. The type of interdependencies of interest in CRUTIAL rests mainly on fault/failure propagation. In particular we have considered the definition of Rinaldi as stated in Section 2.

The aim of our investigation is to find an adequate formal settings in which the three types of failures can be adequately and precisely described. We have posed the following ideal requirements on this formal setting:

R1. It has to be compositional (each infrastructure should be modelled from its own "point of view") – this choice allows separation of concerns and reusability of models in different context (e.g. the model of II could be used for studying the interdependencies between the EI and II, but also between the gas distribution infrastructure and the II);
R2. There should be a notion of "severity" of the state or of an event (as in the definition of escalating failure);
R3. It should be able to model dependencies, and chains of "cause-effect" (needed in all three definitions of failure), and this cause-effect chain can potentially be between states, or events, or a mix of the two;
R4. It should include as few concepts as possible;
R5. It has to be easy to understand also for domain non-expert (that may not be computer scientists)

Typically R4 and R5 are in opposition, since few concepts favour the formal reasoning and the proof, but can make the modelling activity quite cumbersome.

The formal setting we have devised consists of a network of state automata: each infrastructure is modelled as a single automata, and the whole system is described by the composition of the automata.
The states of the automata are labelled with set of propositions, like, for example, {up, down, failed, partially-failed}, and appropriately decorated to account for severity.
Transitions in the automata (edges) are decorated with actions.
The change of state in one automata has been enriched to model both endogenous causes (the action associated to the arc), and exogenous ones (when a state, or an event, in one automata can take place or not depending on the state of another automata, and a change of state of one automata can provoke a change of state in another automata).
Before introducing the class of automata considered (called Dependent Automata, or DA for short), we discuss the problem of dependency in a very abstract setting.
The formalism of DA will then be applied to the model of global dependency of Section 3.1, and to a formalization of the three notions of failure listed before. Finally, we discuss why a new state model has been introduced (instead of reusing the many already existing), and we report on the status of the implementation.

3.2.1 The "Dependent Automata" (DA) model

Since our base was the qualitative model introduced in D3, and considerably enriched in this same document (Section 3.1.1), we consider only two infrastructures generically named A and B. From the study of the model in Section 3.1.1, we have observed the following basic behaviours (seen from the viewpoint of A). A cause of a modification of the system can be:local(A): a state transition in A depends on A only

- depend(A,B): a state transition in A can depend on B being in a well defined state
- Synch(A,B): A and B evolve through the same common action.

The effect of a cause can be:

- local (on A only), or
- global (on A and B)

We have combined these behaviours into Table 4, and we get the following classification (seen from A viewpoint):

- LocalCause, Local effect (LC-LE): when a change of state in A is triggered by an event of A itself, and its effect is confined in A,
- LocalCause, Global effect (LC-GE): when a change of state in A is triggered by an event of A itself, and its effect provokes also change of state in B

- GlobalStateCause, Local effect (GSC-LE): when a change of state in A depends on the state of B, but its effect is confined in A
- GlobalStateCause, Global effect (GSC-GE): when a change of state in A depends on the state of B, and its effect provokes also change of state in B
- GlobalActionCause (GAC): when there is an event that is common to the two infrastructures (or *it is seen* as common in the two infrastructures).

Table 4: the cause-effect relationships

cause* \ *effect	**A only**	**A and B**
local(A)	LC-LE	LC-GE
dependent(A,B)	GSC-LE	GSC-GE
Synch(A,B)	-	GAC-GE

Let us now introduce Dependent Automata, that we define, in this deliverable, only for the case of two automata. Let A and B be such automata.

Definition. An automata A, dependent upon the set of states S_B of an automata B is defined by the tuple:

$$A = (S_A, s_A, E_A, SE_A, L_A)$$

where

- S_A is the non empty and finite set of states
- $s_A \in S_A$ is the initial state
- $E_A \subseteq S_A \times S_A \times Act_A \times P(S_B \times S_B)$, where Act_A is the set of Actions of the automata A, and $P(S_B \times S_B)$ is the power set over the pairs of states of B
- SE_A: $S_A \rightarrow [0,1]$ is the severity function of states, with 1 being the greatest severity
- LA: $S_A \rightarrow \mathcal{P}(AP_A)$ is the labelling of states over a set of symbols AP_A

A labelling of states allows a simple way to classify states in classes (for example AP_A could be {up, down} and L_A (s)=up means that in state s the system is working correctly), as does the set of actions Act_A, that allows to distinguish events from changes of state (different change of state in the automata can actually be due to the same action).

We depict an arc as in Figure 10.

(a) --------- α,*effect*------> (a')

Figure 10: An arc in a DA

where *effect* $\subseteq P(S_B \times S_B)$, we further define *from*(effect) (and *to*(effect)) the set of states that appear as first (second) element of the pairs in *effect*.

When considered in isolation, the meaning of the arc in Figure 10 is that, upon action α, automata A can move from a to a'. When considered in the parallel composition with B, the informal meaning is that, upon action α, automata A can move from a to a', if, at the same time, automata B is in a state b$\in$ *from*(effect). As a consequence of the change of state in A, also B will move from b to b', for (b,b')$\in$ *effect*. This semantics "implements" a concept very similar to "test&set": an action α of A can take place only if B is in a given state (test) and it realization modifies the state (set), that is to say, the value that has been tested.

The formal semantics of (network of) dependent automata is defined through the parallel operator ||, as follows.

Definition. Given two Dependent Automata A = (S_A, s_A, E_A, SE_A, L_A), depending on a set of states S_B of an automata B, and B = (S_B, s_B, E_B, SE_B, L_B), depending on a set of states S_A of automata A, we define the Network of DA (NDA), over the set Synch $\subseteq Act_A \cup Act_B$ the automata defined as follow:

$$Sys = A ||_{Synch} B$$

where Sys = (S, s, E, SE, L), with

- $S = S_A X S_B$ is the set of states
- (s_A, s_B) is the initial state
- $E \subseteq S \times S \times Act$, and there is an arc (a,b) ---α--> (a', b'), if
  - $\alpha \notin$ Synch and $\exists$ (a) --- α,*effect*--> (a') $\in E_A$, with (b,b')$\in$ *effect* OR
  - $\alpha \notin$ Synch and $\exists$ (b) --- α,*effect*--> (b') $\in E_B$, with (a,a')$\in$ *effect* OR
  - $\alpha \in$ Synch, and $\exists$ (a) --- α,Φ--> (a') $\in E_A$, and $\exists$ (b) --- α,Φ--> (b')) $\in E_B$. where Φ is the empty set.
- SE: S $\rightarrow$ [0,1] with SE(a,b) = min{SE(a), SE(b)}
- L: S $\rightarrow$ $AP_A \times AP_B$., with L(a,b) = (L(a),L(b))

The rule for synchronization can be extended to consider the case of non empty effects, by considering the following extended rule

- $\alpha \in$ Synch, and $\exists$ (a) --- α,$effect_i$--> (a') $\in E_i$, and $\exists$ (b) --- α, $effect_j$ --> (b')$\in E_j$. with (b,b')$\in$ $effect_i$ and (a,a') $\in$ $effect_j$

Note that:

- if $\exists$[b, b'] $\in$*effect* and [b, b"]$\in$*effect,* that is to say $\exists$ b: |*effect* (b)| > 1, then there is a non deterministic effect of (a --->a') over B
- if *effect* is empty, A cannot evolve from a
- if *effect* = Id, with Id = $U_{b \in S_B}$ [b, b], then A can evolve independently of B.As for their name, by default the behaviour of a DA depends upon the behaviour of the other automata. To favour the definition of automata in isolation, that is to say when S_B is unknown or may vary in time, we can enrich the syntax of the effect function to allow for an identity functions (pairs of equal states) for any state of S_B, even if S_B is not fully known, and an identity-complementary functions, that adds identity pairs for all whose elements of S_B that are not included in *from(effect)*.

We can therefore extend the definition of *effect* as follows:
effect = effect' $\cup Id_B \cup Compl_B$(effect')
where, referring to B:

- $Id_B = \bigcup_{b \in S_B}$ [b, b])
- *Compl*(effect') = {(b, b), for all b| no pair [b, b'] $\in$*effect'*}, that can be also defined as: *Compl*= Id_B \ *from*(*effect'*).

The definitions are given in the context of A (so the effect is defined on the states of B), but same applies for B.

From a behavioural point of view, the integration of Id_B into the effect of an arc from a to a' ensures that the automata can take the arc independently of the state of the other automata, while the integration of the *Compl* function also ensures that the automata can always follow

the arc, but not necessarily in an independent manner. It is obvious that if *effect* is the empty set, then *Compl= Id.*

Let us now discuss how the 4 combinations of LC/GSC causes with the LE/GE effects given in Table 4 can be modelled in DA. An arc from a to a' labelled (α, *effect*) represents an instance of one of the four situations as follows:

- LC-LE: if $\alpha \notin$ Synch and *effect=Id*,
- GSC-LE: if $\alpha \notin$ Synch and effect = Id' $\subseteq$Id; indeed the automata can follow the arc only if the other automata is in a given subset of states; the change of state induced by the arc as no effect on the other automata
- LC-GE: if $\alpha \notin$ Synch and *effect* is a total relation and *effect* $\neq$Id; indeed the automata can take the arc independently of the state of the other automata (since *effect* is a total function); the change of state induced by the arc has an effect on the other automata (since effect is different from Id)GSC-GE: if $\alpha \notin$ Synch and effect is characterized by *From*(*effect*) $\subset S_B$ (effect is not a total function) and $\exists$ [b, b'] $\in$*effect* with b$\neq$b'
- GAC-GE: if $\alpha \in$ Synch

Note that the definitions above are mutually exclusive, and therefore an arc in a DA can always be classified as either LC-LE, LC-GE, GSC-LE, GSC-GE, or GAC-GE.

Let us consider the five requirements that we have listed at the beginning of this section, so as to discuss to what extend, and how, the introduced settings provide an answer to the listed requirements

R1. *It has to be compositional*: this is not compositional in the classical process algebra sense, because not only the action names are relevant, as in any process algebra, but also the states names of the other automata. As a consequence, a change in a name of a state of one infrastructure has to be reflected in all other infrastructure's models that referred to that state – not quite the classical meaning of compositional. The use of functions *Id* and *Compl* can limit the impact of successive changes.

R2. *The notion of Severity* is introduced but it is not really exploited: the min function is a very conservative approach that could be articulated in different variations, if needed.

R3. *It should be able to model dependencies, and chains of "cause-effect" (needed in all three definitions of failure), and this cause-effect chain can potentially be between states, or events, or a mix of the two*: we think that this objective has been fulfilled, as demonstrated by the fact that the basic classes of dependencies can be clearly defined in DA.

R4. *Minimality of concepts*: we have considered the whole set of dependencies through two basic rules, synchronization and effect function, and by playing with the effect function we can describe quite different forms of dependency and independency. The use of state label and severity function may be seen as redundant: it would be enough to define a total order on the set of state label, saying, for example, that fail>error, but we found it too un-natural and less flexible.

R5. *It has to be easy to understand*: more investigation and experimental evidence should be gathered on this point. Clearly the point of view is different from the classical composition of automata, since the focus is on the dependencies between automata and not on the local behaviour: what we have is a language in which describing dependency comes at little cost, while describing independent behaviour is a little more elaborated. Readability should also be improved, especially in the implementation, for example by distinguishing the arcs a-priori into the 5 dependency classes, to then enforce appropriate definitions of the *effect* function associated to the arc.

3.2.2 Application to the qualitative model

The qualitative model proposed in D3 and enriched in Section 3.1.1, is a very good example for the formalism, since most of the actions considered relate to the dependencies between two infrastructures: EI and II, and there is little (not to say none) local behaviour. Figure 11 is one of the qualitative models contained in D3, in particular it is the one published in [Laprie *et al.* 2007] : totally the automata of the whole system has 8 states and 22 arcs (9 represent EI actions, 9 represent II actions, and 4 are synchronizations)

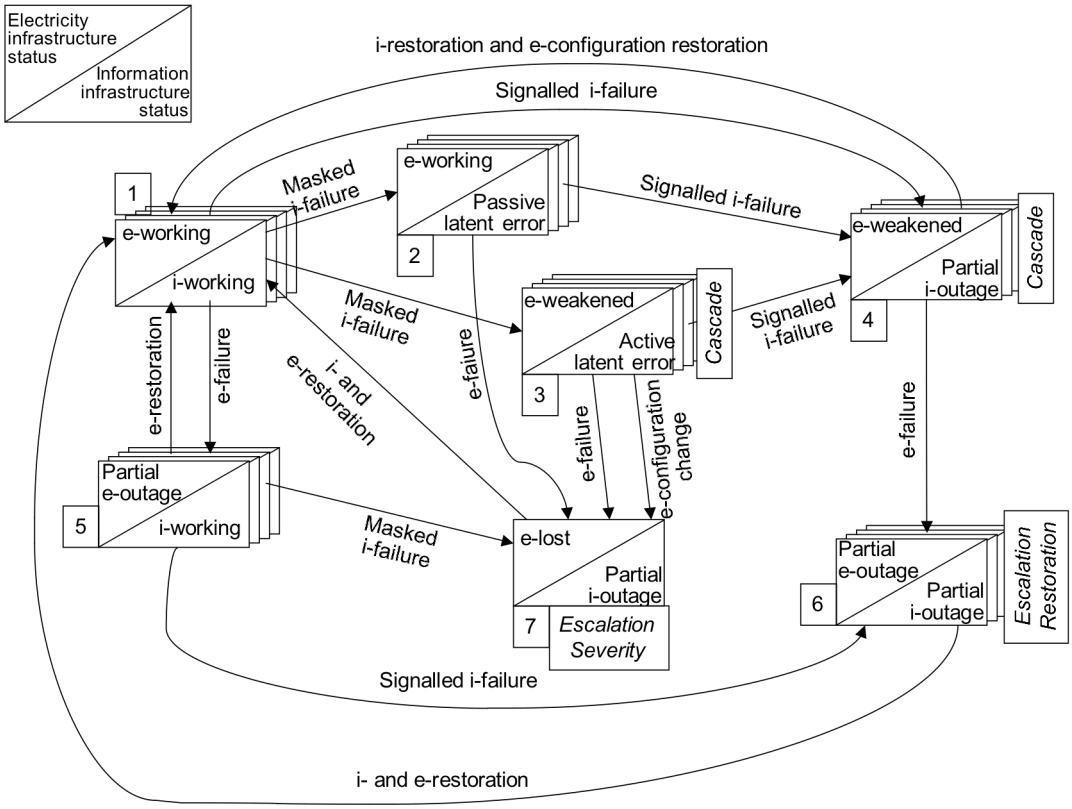


Figure 11: The qualitative model of D3 considered

Table 5 and Table 6 show, in textual form, the II automata as a list of the elements of E; each line in the table is an arc, in the order: the identifier of the arc in the model of D3, the *from()* and *to()* states of the arc, the action that labels the arc, and the pairs that constitute the *effect* function. Table 7 and Table 8 report the same information but for the EI infrastructure.

Let II be the DA of the ICT infrastructure, and EI be the model of the electrical infrastructure, then the global model is given by

$$\mathbf{Sys} = \mathbf{II} \;||_{\{\text{e- and i-restoration}\}}\; \mathbf{EI}$$

Table 5- States of the II DA

States	Label	SE()
i-work	i-work	0
pass-err	pass-err	0.6
act-err	act-err	0.4
i-outage	fail	1
i-weak	i-weak	0.2

Table 6- Arcs of the II DA

id.	s	s'	ACTION	EFFECT1	EFFECT2	EFFECT3	Dependency type
1	i-working	passive latent error	masked i-failure passive	e-working, e-working			GSC-LE
2	i-working	i-outage	masked i-failure	partial e-outage, e-lost			GSC-GE
3	i-working	i-outage	signalled-i-fail	e-working, e-weakened	partial e-outage, partial e-outage		GSC-GE
4	i-working	active latent error	masked i-failure active	e-working, e-weakened			GSC-GE
5	passive latent error	partial i-outage	signalled-i-fail	e-working, e-weakened			GSC-GE
6	active latent error	partial i-outage	signalled-i-fail	e-weakened, e-weakened			GSC-GE
7/8	partial i-outage	partial i-outage	e- and I-restoration	e-lost, e-working	partial e-outage, e-work	e-weakened, e-working	GAC
10	i-weakened	partial i-outage	masked i-failure	partial e-outage, e-lost			GSC-LE
9	i-weakened	partial i-outage	signalled-i-fail	partial e-outage, partial e-outage			GSC-LE
11, /12	i-weakened	i-working	e- and i-restoration	partial e-outage, partial e-outage			GAC

Table 7- States of the EI DA

States	Label	SE()
e-working	e-work	0
e-lost	fail	1
partial e-outage	fail	0.8
e-weakened	e-weak	0.2

Table 8- Arcs of the EI DA.

id.	s	s'	ACTION	EFFECT1	EFFECT2	Dependency type
a	e-working	partial e-outage	e-failure	i-working, i-weakened	i-working, i-working	GSC-GE
b	e-working	e-lost	e-failure	passive latent error, i-outage		GSC-GE
d	partial e-outage	partial e-outage	e-failure	i-working, i-weakened		GSC-GE
d'	partial e-outage	partial e-outage	e- and i-restoration	i-weakened, i-working		Synch
c	partial e-outage	e-working	e- restoration	i-working, i-working		GSC-LE
e	partial e-outage	e-working	e- and i-restoration	partial i-outage, i-working		Synch
f	partial e-outage	e-lost	e-failure	partial i-outage, partial i-outage		GSC-LE
g/m	e-weakened	e-working	e- and i-restoration	partial i-outage, i-working		Synch
h	e-weakened	partial e-outage	e-failure	partial i-outage, partial i-outage		GSC-LE
i	e-weakened	e-lost	e-failure	active latent error, partial i-outage		GSC-GE
l	e-weakened	e-lost	e-configuration-change	active latent error, partial i-outage		GSC-GE
n/o	e-lost	e-working	e- and i-restoration	partial i-outage, i-working		Synch

Considering the two tables is quite evident that the two models are tightly coupled. Indeed the EI model has four different states, and none of the change of states in the II is independent from the state of EI (note that a GSC may not represent a dependency in the "infrastructure" sense of the word, but it may simply reflect the choice of the modeller to restrict the description of the ICT to its behaviour with respect to the EI behaviour), moreover many effect of II are of the GE type (that is to say the modeller has mainly concentrated on the II transitions that do cause a change of state in the global system).

From such a description of the system it is rather straightforward to produce the model of the influence of EI over II and vice-versa (that were present in the model construction process of the unified models of D3), by projection of the table, taking into consideration the *effect* function (Table 9 and Table 10). Indeed the *effect* function plays the role that in the development of the unified models is played by the automata that describes the influence of one infrastructure over the other.

Table 9- Impact of EI over II

II states		**EI actions**
s	s'	action
active latent error	partial i-outage	e-failure
active latent error	partial i-outage	e-configuration change
partial i-outage	partial i-outage	e-failule
partial i-outage	i-working	e- and i-restoration
i-weakened	i-working	e- and i-restoration
i-working	i-working	e- restoration
i-working	i-weakened	e-failure
passive latent error	partial i-outage	e-failure

Table 10- Impact of II over EI

EI states		**II actions**
s	s'	s
partial e-outage	e-lost	partial e-outage
partial e-outage	partial e-outage	partial e-outage
partial e-outage	partial e-outage	partial e-outage
partial e-outage	e-working	partial e-outage
e-weakened	e-weakened	e-weakened
e-weakened	e-working	e-weakened
e-working	e-weakened	e-working
e-working	e-weakened	e-working
e-working	e-working	e-working
e-lost	e-working	e-lost

3.2.3 Modelling interdependencies in the DA model

When do dependencies as discussed in the previous sections do actually represent failures? We now discuss how the dependency expressed in a DA model can help in providing a formal setting for the failure definitions given in the beginning of Section 3.2.

What is a failure, and how can a failure be represented inside a state machine? Let us consider various possibilities:

- failure modelled by an action name; to avoid the detail of having to identify and distinguish fault, error, and failure, the term *disruption* is used to refer to a failure event,failure modelled by a triplet (from_state, action, to_state): corresponds to a *fail transition,* failure modelled by a state reached upon an event: this corresponds to *a fail state,*failure modelled by the label of the state reached upon an event: this corresponds to a failure label attached to states, and it is a generalization of the previous one,failure modelled by the SE function of a state: a state is a failure if its

SE function is above a given threshold (not considered any longer since there is no real advantage over using the state labels. To proceed further, we need to define two sets: fail actions Fail-act⊆Act and fail states Fail-state⊆AP and fail states. We can the define:

Definition. A state *a* is a failure state if L(a) ∈Fail-state

Definition. An action *a* is a *failure action* if it belongs to a subset Fail-act ⊆ Act

Definition. An action *a* is a *potential failure* if there is at least a triplet (*from-state, α, to-state*), such that L(*from-state*)∉Fail-state and L(*to-state*)∈Fail-state

We consider three automata: two DAs A, B and an automata C= A||B.

In the next three paragraphs we formalize different notions of cascading, escalating and common-cause failure.

3.2.3.1 Cascading failure

The definition of cascading failure given in deliverable D3 states that: a *cascading failure* occurs when a disruption in one infrastructure causes the failure of one or more component(s) in a second infrastructure.
The definition of Rinaldi [Rinaldi *et al.* 2001] for *cascading failure* states that a cascading failure occurs when a disruption in one infrastructure causes the failure of a component in a second infrastructure, which subsequently causes a disruption in the second infrastructure.

The two natural language definitions of cascading failure are very similar. The use of the term “disruption” emphasizes that failure is an action. We can instantiate these definitions in two manners: either a failure in A immediately causes B to enter a failure state, or it causes B to move to a state from which a fail action for B can take place, leading B (in 0 or more steps) into a failure state.

Definition. A *0-step cascading failure* of A over B is a transition

$$(a,b) -\alpha \rightarrow (a',b')$$

in A||B, such that L(b)∉Fail-state and L(b’)∈Fail-state. Note that it is not requested that neither L(a) nor L(a’) ∈Fail-state. A 0-step cascading failure of A over B is caused by a *-GE arc in A

Definition. An *n-step cascading failure* of A over B is a sequence of transitions

$$(a^0,b^0) -- \alpha \rightarrow (a^{1'},b^1) --- \beta^1 \rightarrow (a^{1'},b^2)) --- \beta^2 \rightarrow (a^{1'},b^3) \ldots\ldots\ldots --- \beta^{n-1} \rightarrow (a^{1'},b^n)$$

in A||B, such that $\alpha \in Act_A$, for k∈[1..n-1]: $b^k \in Act_B$, for j∈[0..n-1], L(b^j)∉Fail-state and L(b^n)∈Fail-state

Note that what we have in the n-step is a *-GE arc in A followed by a number of LC-LE arcs in B.
Moreover it is assumed that a failure action in A provokes B to reach a fail state, either directly, or through a series of actions, nevertheless there is no direct implication between the action of A that generates the cascading failure and any action in B, since this connection passes through the states of B.

Another possibility to define cascading failure of A over B is to consider whether a failure action of B is conditioned over A being in a failed state, so we can give the following

definition, that we call *conditional cascading failure* (a failure in B is conditioned on A being in a failure state).

Definition. A *conditional cascading failure* of A over B is an action *b* of B such that *b* can take place only if A is in a state *s* which is a failed state (the *effect* associated to action *b* is such that from(effect) $\subseteq$ Fail-state$_A$, that is to say it includes only fail states of A).

The definition above is very close to the concept of escalating failure introduced in the rest of this section.
Note that none of the three definitions establishes a 1:1 connection between a failure in A and a failure in B, so we are considering cases in which a failure in A may provoke a failure in B, but a failure in B is not necessarily the only possible evolution of the global system upon a failure in A.

3.2.3.2 Escalating failure

The definition of escalating failure given in deliverable D3 states that an *escalating failure* occurs when an existing *failure* in one infrastructure exacerbates an independent disruption in another infrastructure, increasing its severity or the time for recovery and restoration from this failure.

The definition of Rinaldi [Rinaldi *et al.* 2001] for *escalating failure* states that an *escalating failure* occurs when an existing *disruption* in one infrastructure exacerbates an independent disruption of a second infrastructure, generally in the form of increasing the severity or the time for recovery or restoration.
We follow the definition given in D3, which states more clearly that there is a failure state (and not an action) and is more precise, but nevertheless it is underspecified whether an escalating failure is a fail action, a fail state, or a path in the system, since it is not said "what it is" but "when it happens".
We consider various cases, assuming that the disruption in the definition happens in A and its effects worsen due to the state of B. Note that in the formalism we have a notion of severity (so failure that escalates in severity can be adequately modelled), but there is no notion of time to fail or to recover, so this aspect cannot be modelled explicitly. In the following list, we always assume that Action $\alpha \in$ Fail$_A$.

1. *Local state differs*. There are two or more states reachable from *a* upon action α, and they differ in severity. The choice of the transition to follow depends upon the state of B
2. *Time to fail differs*. There are two or more transitions from *a* upon action α, possibly leading to the same state *a'*, the time of the associated transition differ, and the choice of the transition to follow depends upon the state of B.
3. *Recovery state is different*. There are two or more recovery states reachable from *a* (where *a* is a Fail state of A reached upon action α. The choice of the transition to follow towards recovery states depends upon the state of B
4. *Recovery action is different*. There are two or more recovery actions from *a* (where *a* is a Fail state of A reached upon action α), possibly leading to the same state. The choice of the transition to follow towards recovery states depends upon the state of B. The different actions could have an associated recovery time that is different.

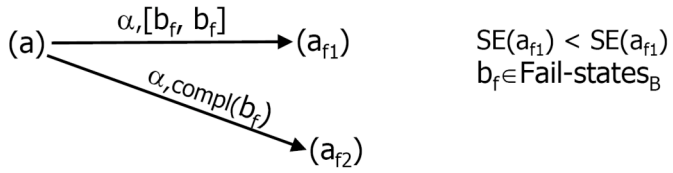


Figure 12: Escalating failure, local states differ

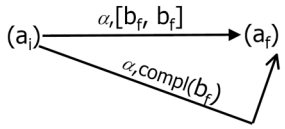


Figure 13: Escalating failure, time to fail differ

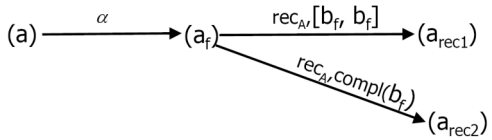


Figure 14: Escalating failure, recovery state is different

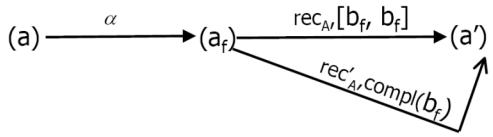


Figure 15: Escalating failure, recovery action is different

3.2.3.3 Common-cause failure

The definition of common-cause failure given in D3 is: a *common cause failure* occurs when two or more infrastructures are affected simultaneously because of some common cause. The definition of Rinaldi [Rinaldi *et al.* 2001] for *common cause failure* states that a common cause failure occurs when two or more infrastructure networks are disrupted at the same time: components within each network fail because of some common cause.

For modelling common cause failure we need an additional system C that plays the role of the *environment*.
An action α in C is a *common cause failure* if it provokes a change of state in both A and B, and the resulting states in A and B $\in$ Fail-state.
Note that a common cause failure is an application of a *-GE rule of a third system on A and B, with an additional condition on the resulting states (second component of *effect* associated to *a*).

For the time being, since the composition of more than two automata has not been defined yet, we cannot define formally, using the DA formalism, what a common cause is

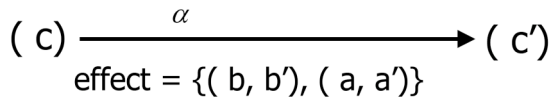


Figure 16: Common-cause failure

3.2.4 Planned extensions

The work will continue along different lines: WP5 will provide an implementation of DA inside the open framework of DrawNET, WP2 will consider the cases of dependencies in the selected scenarios, and will consider a possible extension of DA to deal with apparent states: at the moment it is not totally clear whether the composite state “real/apparent” can be seen as a composition of two DAs, or if an extension of the formalism is required .

3.3 PETRI NETS BASED COMPOSITIONAL MODELLING

3.3.1 Introduction

The goal of this section is to present a compositional approach for representing the behaviours of the II and EI infrastructures highlighting the interdependencies existing between them.

The formalism selected for modelling the two architectures is the Petri Nets (PNs) formalism, which is well suited for modelling Discrete Event Dynamic Systems (DEDS). Its mathematical foundations allow interesting qualitative properties of the modelled systems to be checked (e.g., liveness, boundedness, deadlock-freeness). Moreover the introduction of time (extending the PN formalism) makes PN suitable for performance and dependability analysis purposes. In particular the timed nets that we will use in the next sections are modelled through the Generalized Stochastic Petri Net (GSPN) [Ajmone Marsan *et al.* 1995], where two different types of transitions are available: a)the timed transitions having a non null random delay with negative exponential distribution and b)the immediate transitions having zero delay. We will also discuss a possible extension of the presented models using the Stochastic Well Formed Net (SWN) [Chiola *et al.* 1993], a High Level Petri Net formalism (HLPN) that offers a compact system description associating structured information to the tokens flowing through the model structure and an efficient solution technique that automatically exploits the behavioural symmetries of the model.

The choice of presenting a compositional representation of the EI and II behaviours is justified because the sub-models are more readable than the global model (e.g. it is easier to highlight the interdependences between the infrastructure) and they can be more easily extended to take into account all kinds of dependencies and to refine the models for quantitative evaluation purposes.

This approach. that we will present in details in the next two subsections, consists in two phases:

- the modelling of the behaviour of one infrastructure conditioned on the state of other infrastructure, highlighting the *LC* and *GL* consequences of the events (as defined in Section 3.2), and separating the failure behaviours from the recovery/restoration one.
- the composition of the obtained PN sub-models for generating a single complete PN using superposition over places and transitions.

Let us to observe that the state space of this composed PN is equivalent to the state space of the automaton presented in Deliverable D3 [Kaâniche *et al.* 2007]; in fact it is possible from the Reachability Graph (RG) of this net to automatically deduce the automaton

presented in D3. This justifies our choice of maintaining the same notation for the name of the states and the events of the two infrastructures. In Table 11 and Table 12 the mapping between this notation and the new one adopted in Section 3.1 of this deliverable is shown.

In the rest of this section, we consider as a starting point the model presented in D3, that is reminded in Figure 17.

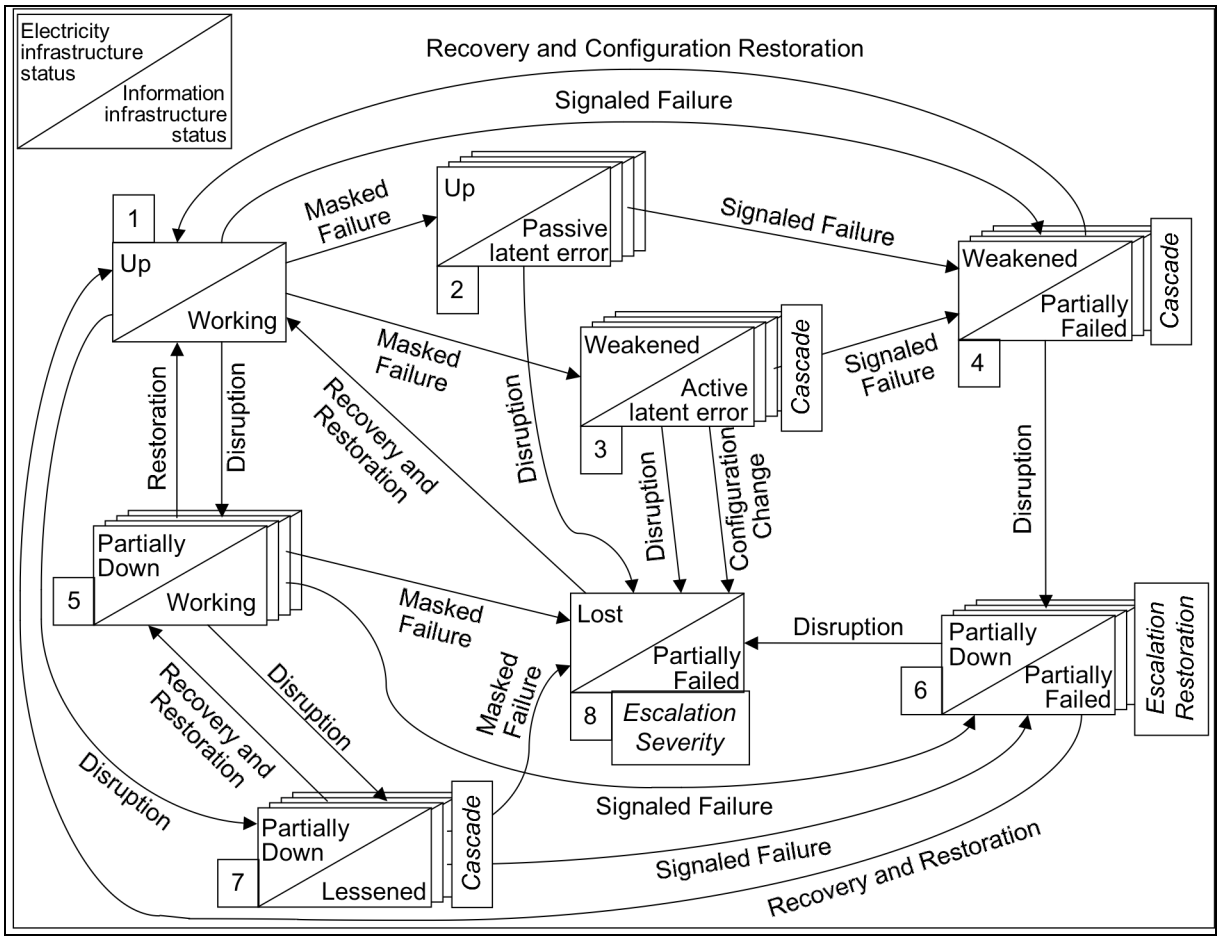


Figure 17: State machine model of the two infrastructures considered as a starting point for model composition (from Deliverable D3)

In Table 11 and Table 12 the mapping between notation used in Figure 17 and the new one adopted in Section 3.1 of this deliverable is shown.

Table 11: Mapping between the old notation and the new notation used for the states of the two architectures

Old notation used for the state of the two architecture	New notation used for the states of the two architectures
EI	
UP	*e-working*
Weakened	*e-weakened*
Partially Down	*partial e-outage*
Lost	*e-lost*
II	
Working	*i-working*
Passive latent error	*passive latent error*
Active latent error	*active latent error*
Partially Failed	*partial i-outage*

Table 12: Mapping between the old notation and the new notation used for the events of the two architectures

Old notation used for the events of the two architecture	New notation used for the events of the two architectures
EI	
Disruption	*e- failure*
Restoration	*e-restoration*
II	
Working	*i-working*
Masked Failure	*unsignalled i-failure*
Signalled Failure	*signalled i-failure*
Configuration Change	*i-configuration change*
Recovery	*i-restoration*

3.3.2 Model description

In this section we present the PN models. Two different modelling approaches are adopted: the former for modelling the failure behaviours and the latter for modelling recovery and restoration behaviours. In the models two types of transition appear: timed ones (boxes) and immediate ones (black bars). The immediate transitions have higher priority than timed one (since they fire with zero delay after their enabling).

For the first modelling approach, in every figure (see e.g. Figure 18) the sub-model on the left represents the behaviour of one infrastructure conditioned on the state of the other one, shown on the right. Moreover the sub-model on the right shows also the possible effects caused on it by the occurrence of an event of the other infrastructure. This is modelled with

immediate transitions labelled with the same label in the right and left models (e.g. the transitions ***|T2-GL*** and ***|T1-LC*** in Figure 18).

We can observe that an immediate transition (e.g. the transition ***|T1-LC*** in Figure 18) in the left model corresponding to a "self-loop transition" on the right one represents a local event. In order to highlight this distinction between the two types of events we have labelled all the immediate transitions with the tags **LC** and **GL,** so that a transition representing an event with only local consequences will be labelled with **LC,** while a transition representing an event with global consequences will be labelled with **GL**. All the places in input to an immediate transition in the left part, representing a vanishing state for this infrastructure, are drawn with a dashed circle. Instead all other places corresponding to the possible (observable) states of the two infrastructures are summarized in Table 13.

Table 13: List of states of each infrastructure appearing in the models.

STATES	
II	**EI**
Working, Passive latent error, Active latent error, Partially Failed.	*UP, Weakened, Partially Down, Lost.*

Moreover the possible failure events in the two infrastructures, modelled by timed transitions, are summarized in Table 14. For instance in Figure 18 the transitions ***Masked Failure*** and ***Signalled Failure*** are timed and represent failures in the information infrastructure.

Table 14: List of events of the two infrastructures appearing in the models.

FAILURE EVENTS	
II	**EI**
Masked Failure, Signalled Failure, Configuration change	*Disruption*

The input places of each net on the left are drawn with bold black line. For instance the input place for the net on the left in Figure 18 is only ***Working***. This means that we expect the left infrastructure to start in one (and only one) of the initial states, given that the infrastructure on the right is in the state represented by the marked place.

Here we assume that the initial state of the global composed model is the state ***Working*** for the **II** and ***UP*** for the **EI**. This is represented in the nets on the left by a black token in all occurrence of these places. Instead the tokens in the sub-models on the right are not considered during the composition (they are used only to show which is the state that conditions the behaviour of the other infrastructure).

Finally, we can observe that the dashed rectangles in Figure 18 and Figure 20 (sub-model on the left) represent the same sub-model ***SubNet1***. This sub-model is connected with these models through the (input) place ***Working*** and the (output) places ***After Masked Failure*** and ***After Signalled Failure***. In the same way in Figure 21 and Figure 22, (sub-model on the left) the dashed rectangles represent the same sub-model ***SubNet2,*** while in Figure 23 and Figure 24 the same sub-model ***SubNet3***. This sub-model is connected with all these models through the (input) place ***UP*** and the (output) place ***After Disruption***.

Figure 18 shows the **II** behaviour (left part) when the **EI** infrastructure is in the ***UP*** state (right part). Observe that the immediate transitions, as already said, are labelled with **GL** or **LC** in order to distinguish their influence with respect to the other infrastructure. For instance we can distinguish between two different types of ***Masked failure*** as shown in Table 15.

Table 15: The two different types of Masked failure

System state before ***Masked failure***		System state after ***Masked failure***	
EI	**II**	**EI**	**II**
UP	Working	UP	Passive latent error
UP	Working	Weakened	Active latent error

The former type of ***Masked failure*** has only a local influence on the **II** (only the **II** state is changed), while the latter one has an influence on both infrastructures (both the **II** and **EI** states change).

Figure 19 shows the **II** behaviour (left part) when the **EI** infrastructure is in the ***Weakened*** state (right part). We can observe that the ***Configuration change*** induces also a change in the **EI** state (from ***Weakened*** to ***Lost***), while the ***Signalled failure*** has only a local effect.

Figure 20 shows the **II** behaviour (left part) when the **EI** infrastructure is in ***Partially Down*** state (right part).

Figure 21 shows the **EI** behaviour (left part) when the **II** infrastructure is in the ***Working*** state (right part).

Figure 22 shows the **EI** behaviour (left part) when the **II** infrastructure is in the ***Passive latent error*** state (right part) .

The two ***Disruption*** events represented in these two last models have different behaviours with respect to the **II.** In the former the ***Disruption*** has only a local consequence on the **EI**, while in the latter has also a side effect on the **II**.

Figure 23 shows the **EI** behaviour (left part) when the **II** infrastructure is in the ***Active latent error*** state (right part).

Figure 24 shows the **EI** behaviour (left part) when the **II** infrastructure is in the **Partially failed** (right part)

PN model describing the **II** behaviour when the **EI** is in the ***LOST*** is not modelled because in this case only ***recovery and restoration*** events are possible.

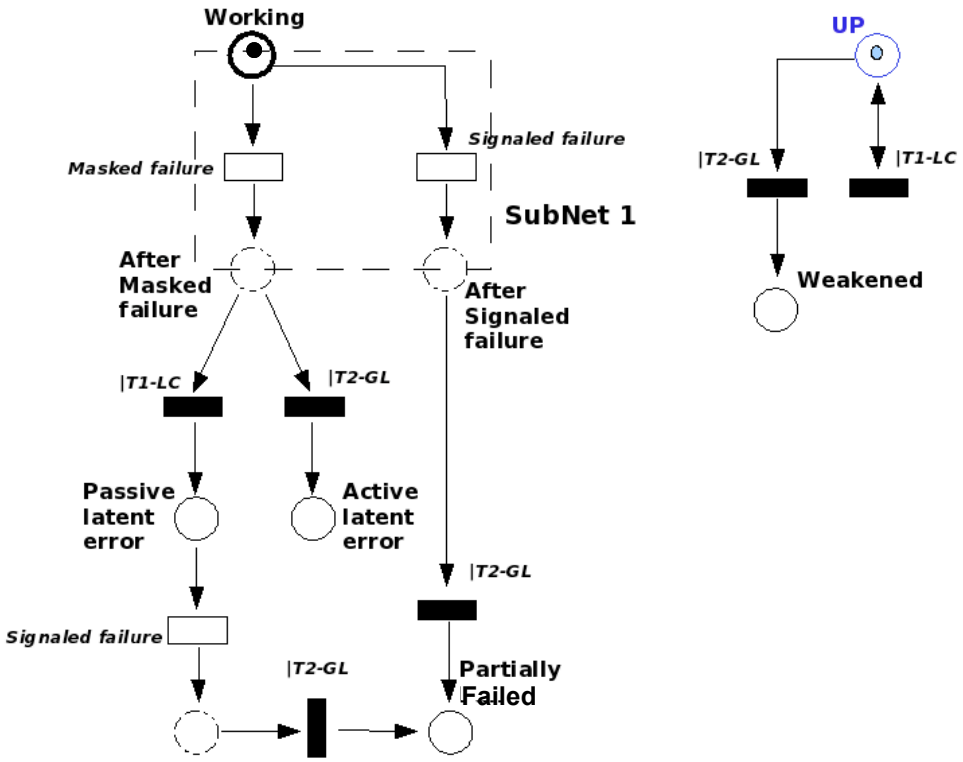


Figure 18: PN showing the II behaviour when the EI infrastructure is in the UP state.

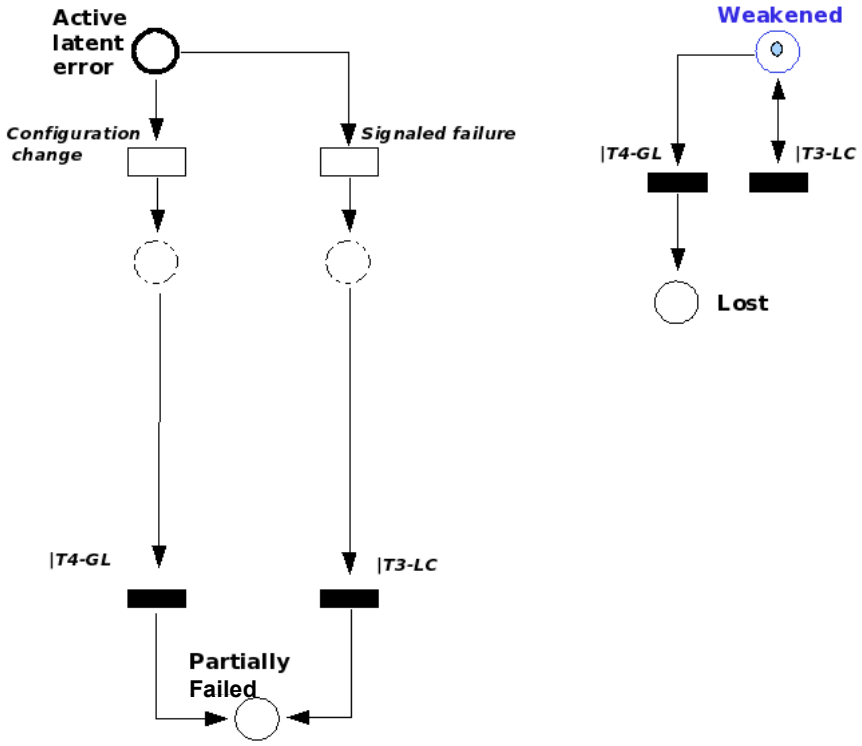


Figure 19: PN showing the II behaviour when the EI infrastructure is in the Weakened state.

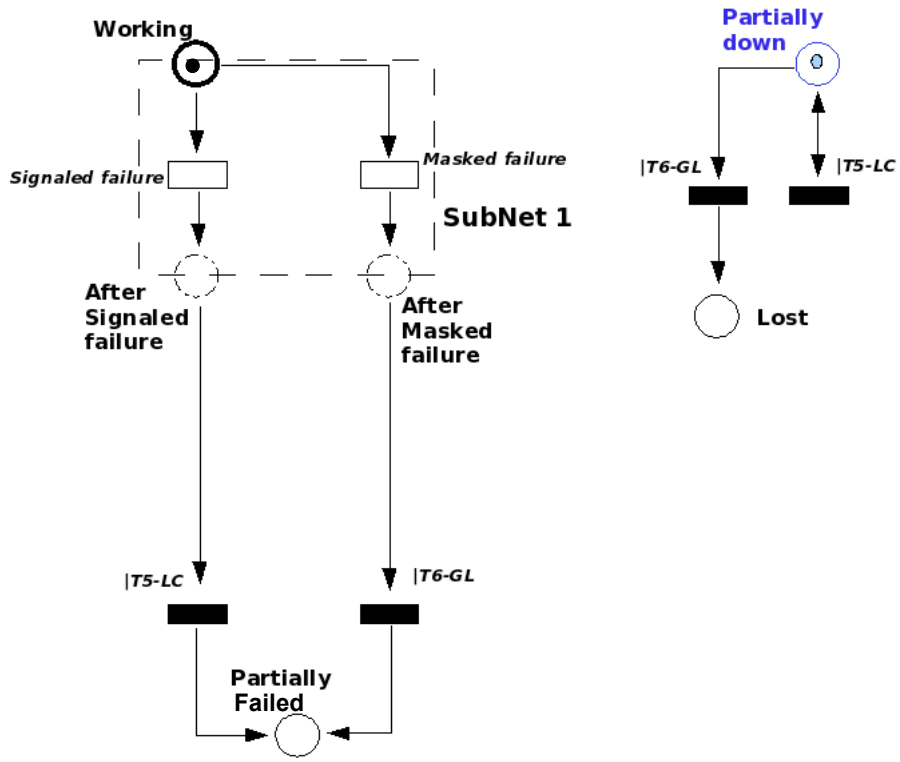


Figure 20: PN showing the II behaviour when the EI infrastructure is in the Partially Down State.

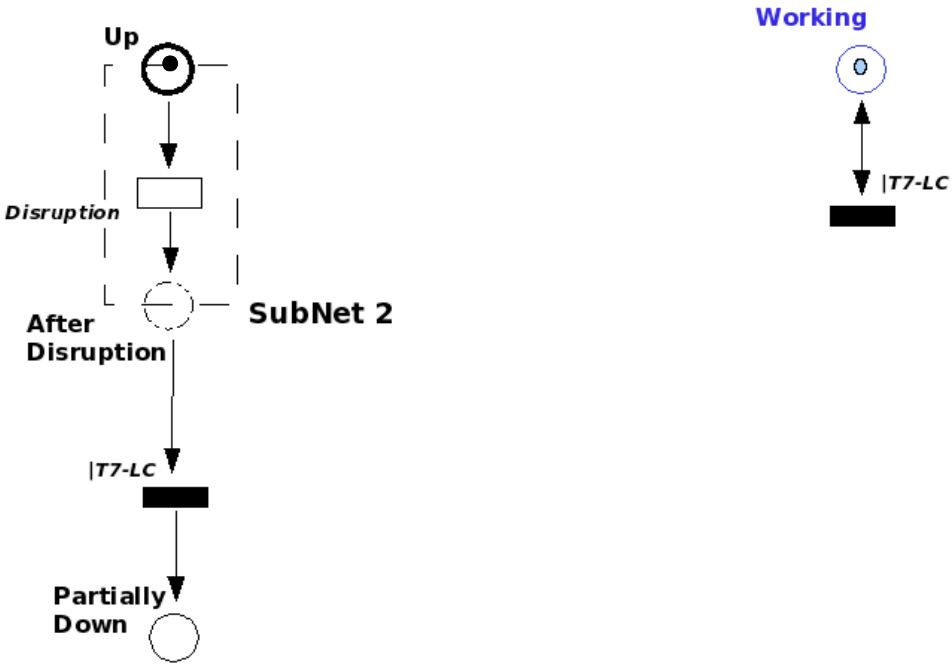


Figure 21: PN showing the EI behaviour when the II infrastructure is in the Working state.

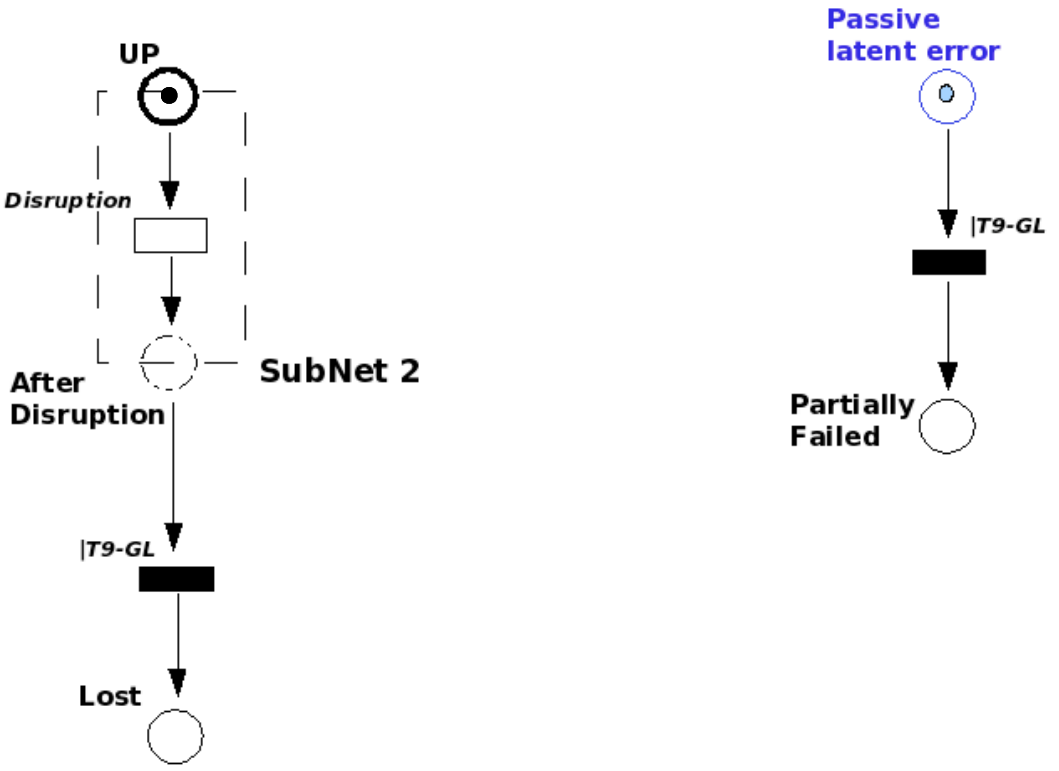


Figure 22: PN showing the EI behaviour when the II infrastructure is in the Passive latent error.

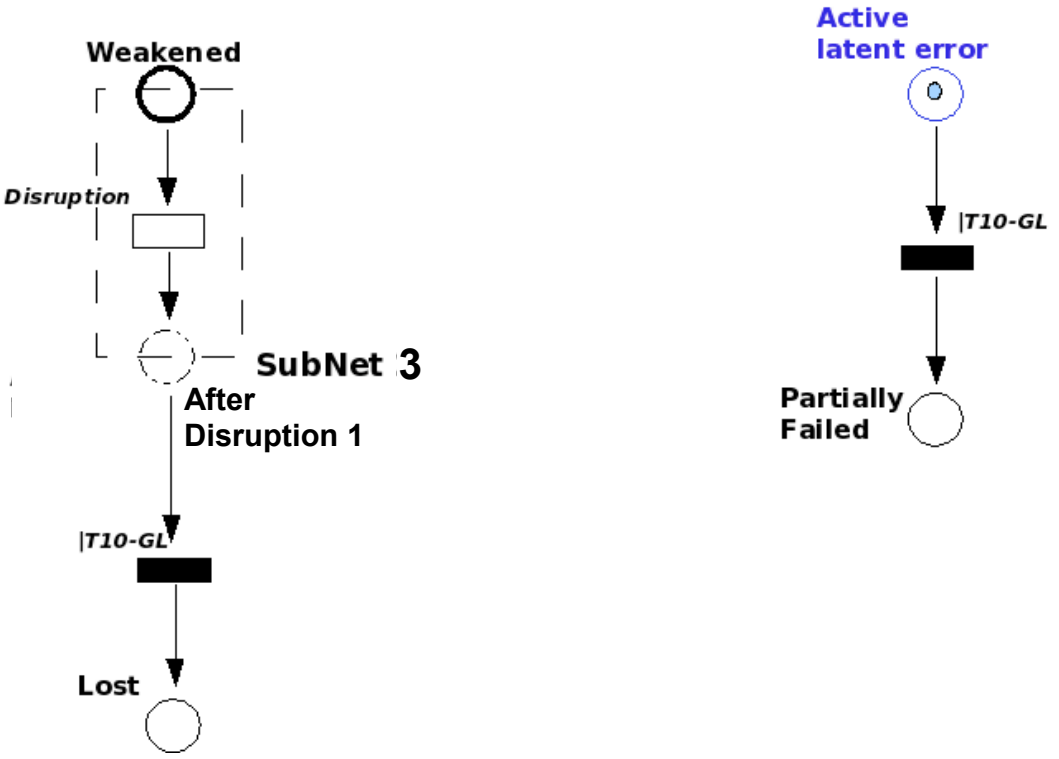


Figure 23: PN showing the EI behaviour when the II infrastructure is in the Active latent error state.

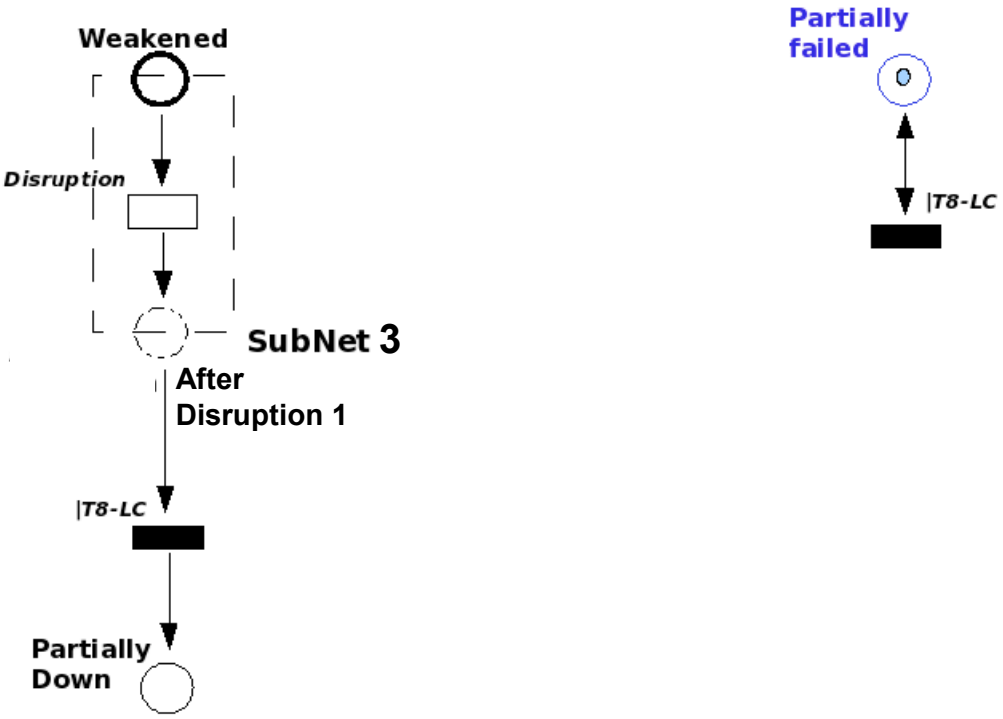


Figure 24: PN showing the EI behaviour when the II infrastructure is in the Partially failed state

The second modelling approach considers the ***recovery and restoration*** events as a unique synchronized event changing simultaneously the state of the two infrastructures, so that a relationship of cause and effect between them does not exist. In fact the ***recovery and restoration*** requires cooperation on both sides. Under this assumption the four PN models representing the recovery and restoration activities are shown in Figure 25.

The possible recovery and restoration events, modelled by timed transitions, are summarized in Table 16, while the set of the places is the same introduced in Table 13.

Table 16: List of recovery and restoration events appearing in the models

Recovery and restoration events
Restoration, Recovery and restoration

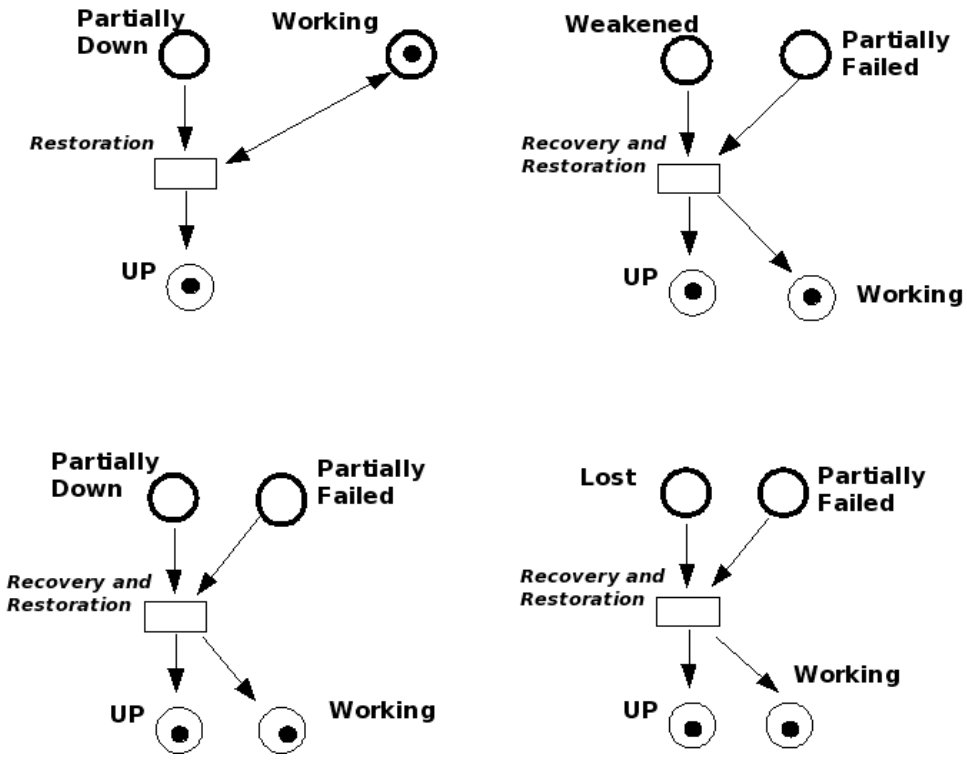


Figure 25: PN showing recovery and restoration events

3.3.3 Composition phase

In this subsection we describe how to compose the sub-models. A single complete net can be derived from these PNs using a composition operator performing superposition over the places and the transitions. Concerning the failure models, the two PNs of every figure can be composed using superposition over the immediate transitions with the same label. Observe that in this phase the sub-models **SubNet1** and **SubNet2** are not considered.

After that, all these models plus the two sub-models **SubNet1** and **SubNet2** and the five sub-models in Figure 25 are composed together by superposition over the places (with the same name) obtaining a unique global PN.

Observe that the Reachability Graph (RG) of this composed PN is isomorphic to the automaton presented in Figure 17[1].

3.3.4 An extension of the previous model

Here we are going to give an example showing how this compositional approach can ease the task of extending or refining the model.

In particular we are going to show how to extend the above PNs for modelling multiple **Disruptions**. This requires to extend only a sub-set of the previous models:

- the PN model showing the II behaviour when the EI is in the Partially Down state;
- the PN model showing the EI behaviour when the II is in the Partially failed state;
- the PN model showing the EI behaviour when the II is in the Working state

[1] The RG, in this context, is the “Tangible Reachability Graph” including only the tangible states.

and to introduce two new models:

- the PN model showing the EI behaviour when the II is in the Lessened state;
- the PN model showing the recovery restoration activity when the system is in the state Partially Down/Lessened

Observe the set of possible states of the two architectures is extended with the state ***Lessened*** as shown in Table 17, while the set of the possible failure event for each infrastructure is the same already introduced and summarized in Table 11.

Table 17: Extended list of states w.r.t. the two infrastructures.

STATES	
II	**EI**
Working, Passive latent error, Active latent error, Partially Failed, Lessened	***UP, Weakened, Partially Down, Lost***

The new obtained PNs are shown in Figure 26. Figure 27, Figure 28, Figure 29 and Figure 30.

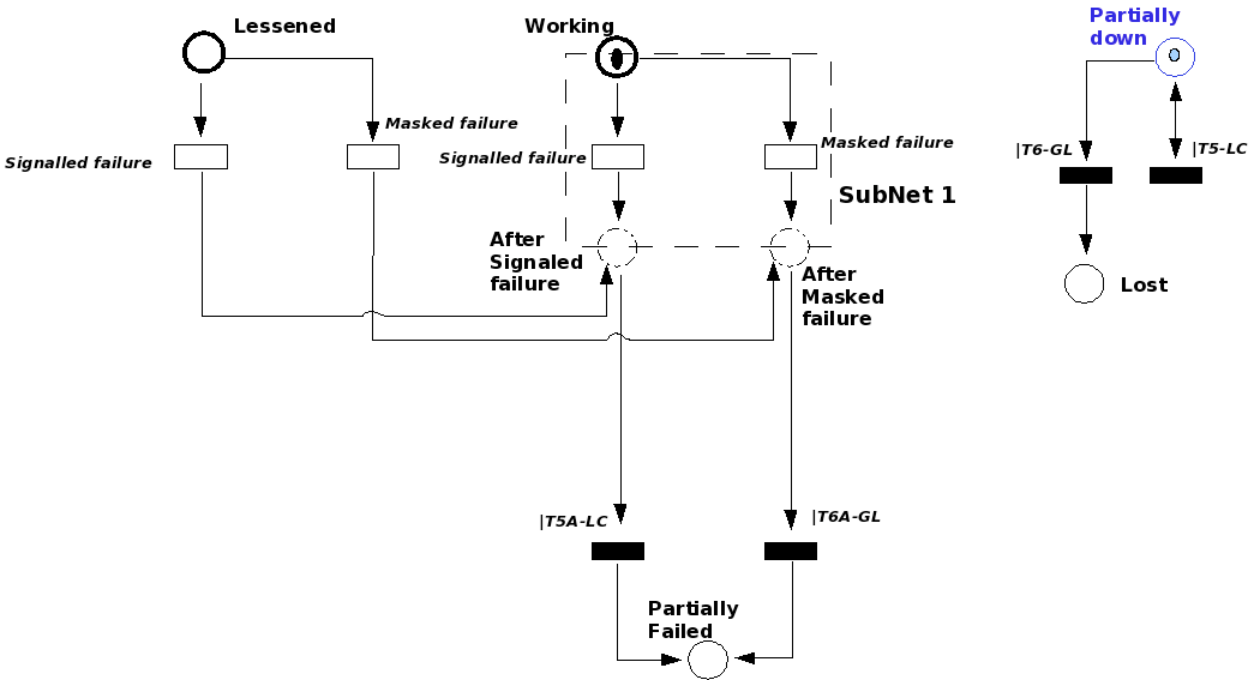


Figure 26: Extended PN model showing the II behaviour when EI is in the Partially Down State.

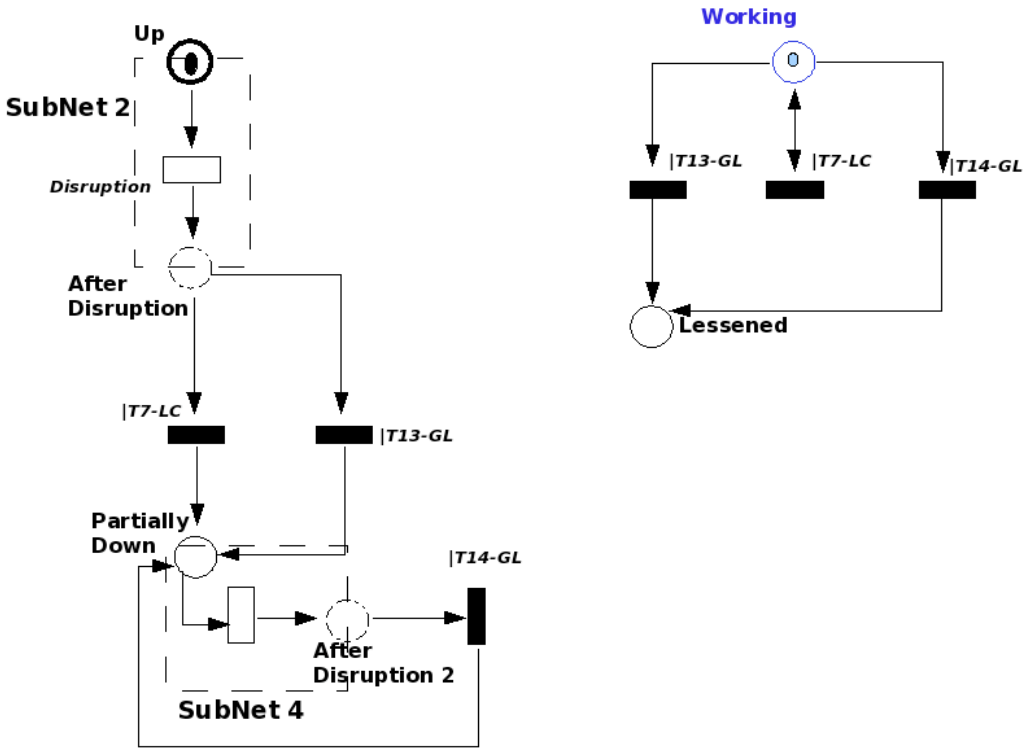


Figure 27: Extended PN model showing the EI behaviour when II is in the Working state

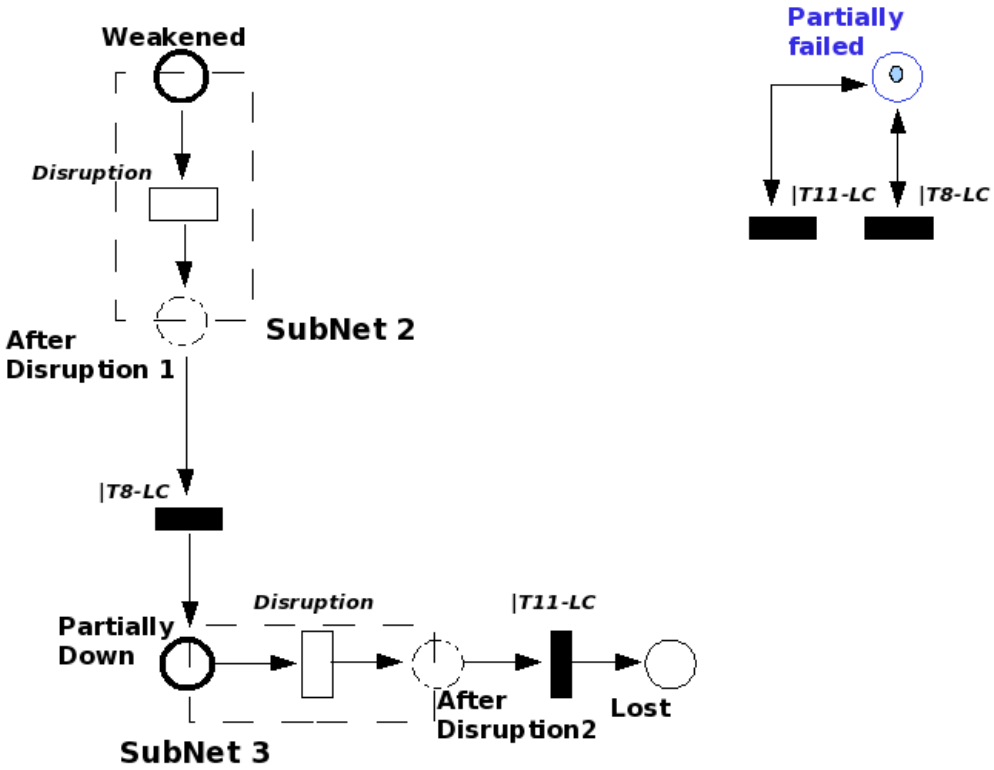


Figure 28: Extended PN model showing the EI behaviour when II is in the Partially failed state

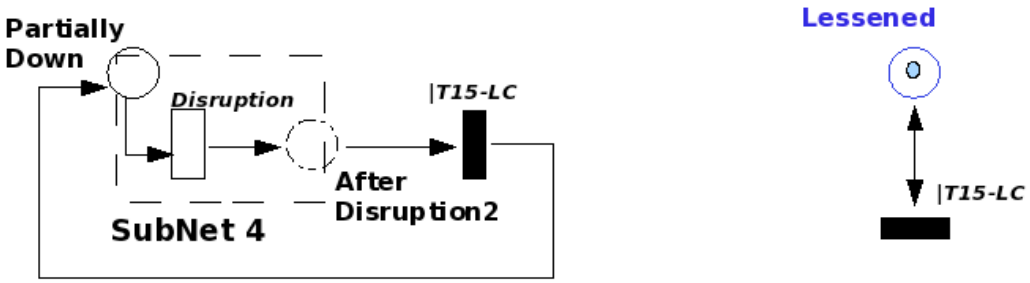


Figure 29: Extended PN model showing the EI behaviour when II is in the lessened state

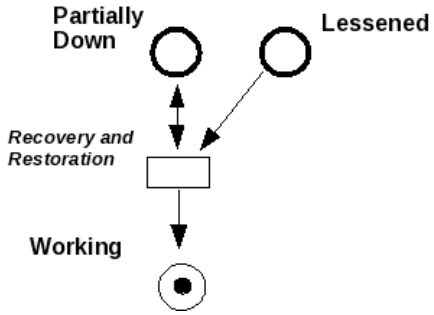


Figure 30: PN model showing the recovery restoration activity when the system is in the state Partially Down/Lessened

3.3.5 Some thoughts on dependability evaluation

The models presented in Section 3.3 can be used as a basis for evaluating dependability measures such as the expected time to reach the state "***Lost/Partially Failed***" from the normal state "***Up/Working***").
First of all we must introduce, in the previous PN models, the temporal concept distinguishing between timed (associated with firing delay) and immediate transitions (firing in zero time) [Ajmone Marsan *et al.* 1995].

In practice we will consider all the low priority transitions (represented by a box) as timed transitions and all the high priority transitions (represented by a black bar) as immediate. To complete the model specification we have to associate a firing delay probability distribution with every timed transition so that it will be possible to compute the performance measures of the model using the standard analysis techniques for GSPN: the numerical analysis (if all the timed transitions have negative exponential distribution) or the simulation.

Two possible approaches can be followed:

- if it is possible to define the firing delay probability distribution of each transition at this high abstraction level then the model is immediately ready for analysis;
- if it is necessary for a specific event to increase the detail level then it is possible in our model to replace the corresponding timed transition with a sub-model. For instance the transition corresponding to the disruption event that moves the system from ***Up/Working*** state to the ***Partially Down/Working*** state can be refined in a complex sub-model representing all the possible classes of disruption events depending on different electrical components faults. In this way we will also be able to distinguish between different variants of the ***Partially Down/Working*** state.

Moreover we can replace of PN portion with a more complex sub-model. This can be useful for modelling “multiple” states. A "multiple" state represents a group of states with the same EI and ICT status, and the events existing from the "multiple" state represent a set of events which can reach or leave a state (or a subset of states) of the "multiple" state.
In this case it is important to observe that a sub-model could also be modelled with a **S*WN*** [Chiola *et al.* 1993], so that the colours can be used to represent the different system sub-states. The use of higher-level model can be more convenient, not only for its compactness and readability but also for its significant degree of parameterisation that can be exploited at the analysis level. In particular the SWN formalism allows to represent very concisely systems thanks to the possibility of associating information with tokens and of parameterizing transition firings, and gives also the possibility to derive efficient techniques working directly on this level (for example taking advantage from the intrinsic symmetries of the system) which can reduce the analysis complexity [Chiola *et al.* 1993].

The model presented in this section could also be used for abstracting the behaviour of a more complex PN model, and could act as a controller. The idea consists in the synchronization of the initial state of the complex model with a corresponding state (or subset of states) in the controller (e.g. if the initial state of the complex model represents a state where the **ICT** and the **EI** are working correctly then this state can be synchronized with the state ***UP/Working***) and in the synchronization of the two model events. In this way it should be easier to define some performance indices for the complex model in terms of the non abstract states and it should be possible to use the controller to validate the complex model behaviour.
For instance if a deadlock is found in the (S)RG of this composed model (complex plus controller models) then an error in the complex model is raised. In fact this corresponds to the case where an event can happen in a specific state of the complex model but not in the corresponding state of the controller.

4 HIERARCHICAL QUANTITATIVE MODELLING APPROACH

In this section, we present a preliminary definition of the framework that could be used: 1) to support the detailed modelling of the interdependencies between the Electrical Infrastructure (EI) and its Information Infrastructure which controls the correct operation of EI (II) and, 2) to quantitatively assess the impact of such interdependencies through proper specified metrics,. Starting from the description of the EI and the II, we have preliminarily characterized the state of both infrastructures, their failure modes and their reciprocal interdependencies. Then, we have taken two development directions, which are being investigated in parallel. On one side, we tackled the definition of modelling components, each one representing an entity of the abstracted Electric Power System[2] and/or interdependencies and failure propagation phenomena, to populate a modelling framework as template evaluation building blocks to be used, in proper combinations, to model (theoretically) any specific EPS scenarios. We started defining a few such template models, to be refined and completed during the third year of the project. An approach to compose these models in a hierarchical fashion is also proposed. Therefore, this activity is meant to develop generic reusable models, accounting as much as possible for internal dynamics of the represented entities as well as dependencies among them, and for generic fault and propagation conditions, in accordance with effective metrics defined to quantitatively assess the impact of interdependencies. Such an evaluation framework has big potentialities in terms of usage, but its development is a delicate activity. So, to support the definition of the generic models, we carried on also another activity consisting in the realization of a simulator based on stochastic models of the EPS which are the same as those of the modelling framework but

[2] The Electric Power System, denoted as EPS, is composed of the EI and II infrastructures.

restricted to a limited number of dynamics/behaviours of the involved entities. Currently, a first version of this ad-hoc simulator, called EPSyS, is completed, and through its usage in simple artificial testbeds, precious feedbacks have been gained.

In Deliverable D3, delivered at the end of the first year, preliminary studies of this hierarchical modelling framework were started. Mainly, the logical schemes of the infrastructures EI and II were identified, including their failure behaviours. Also, the major characteristics which the modelling framework should possess to allow the analysis and evaluation of measures representative of interdependencies were sketched. For the sake of completeness and for a better understanding of the follow-up studies carried on during the second here, in the following we include also the developments already presented in D3.

The rest of this section is organized as follows. Section 4.1 contains the main abbreviations used in the presentation of the framework. In Section 4.2 we present the main system elements to be considered in the framework, basing on [CESI RICERCA 2006a] and [CESI RICERCA 2006b]. Then, the concept of state is defined in Section 4.3, both for the Electrical Infrastructure and for the Information Technology based Control System. In particular, a hybrid state is defined for the EI, composed by a discrete part and a continuous one. In Section 4.4 we identify possible failure models, both within the EI and II individually, and considering their interdependencies. Next Section 4.5 briefly sketches the dynamic behaviour of the electrical power system. Section 4.6 proposes some representative measures of interest, while in Section 4.7 we summarize the functionalities required by the modelling framework in terms of modelling power, modelling efficiency and solution power. The steps to construct an overall EPS model are then outlined in Section 4.8. The behaviour of the main considered EPS components is described in Section 4.9 using flow-chart diagrams, to provide an overall vision of how these components relate in a global picture before going in the details of the models. Section 4.10 deals with how the envisaged modelling framework could be made feasible through the tool Mobius. The definition of the ad-hoc simulator EPSyS is contained in Section 4.11, while final discussion is in Section 4.12.

4.1 Main abbreviations

- EPS: Electrical Power System composed of EI and II
- EI: Electrical Infrastructure
- II: Information Infrastructure
- HG - LG: High Voltage - Medium and Low Voltage Generation plants
- TG - DG: Transmission - Distribution Grid
- HL – LL: High Voltage - Medium and Low Voltage Loads
- LCS: Local Control System
- RTS: Regional Telecontrol System
- NTS: National Telecontrol System
- LCC: Local Control Center
- ATC: Area Telecontrol Center

4.2 Logical scheme of the electrical power system

The content of this Section has been derived from [CESI RICERCA 2006a] and [CESI RICERCA 2006b]. The electrical power system (EPS) is logically structured in two interacting parts: Electrical Infrastructure (EI) and the Information Infrastructure (II).

4.2.1 The Electrical Infrastructure

The EI represents the electrical infrastructure necessary to produce and to transport the electrical power towards the final users. It can be logically structured in different components, as shown in Figure 31(a): the transmission grid (TG), the distribution grid (DG), the high voltage generation plants (HG), the medium and low voltage generation plants (LG), the high voltage loads (HL), the medium and low voltage loads (LL).

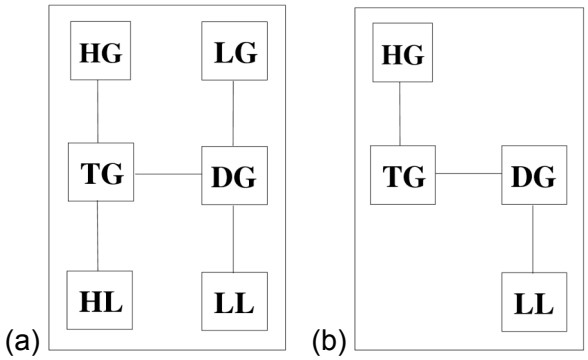


Figure 31: General (a) and typical (b) scheme of the EI

A typical scheme of EI is shown in Figure 31(b) where the components HL and LG are not present. Moreover, the distribution grid can be structured in two different medium and low voltage grids.

From a topological point of view, TG and DG can be considered like a network, or a graph, as shown in the example of Figure 32(a). The nodes of the graph represent the substations, while the arcs represent the power lines. The generators and the loads are nodes connected by arcs (power lines) to the nodes of the grid. Some nodes of the grid can be connected to nodes of the contiguous grid.

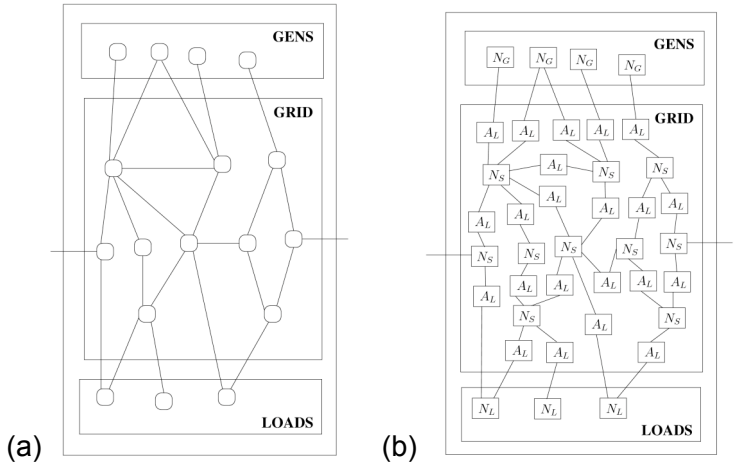


Figure 32: Example of meshed graph (a) and logical graph (b) for a dummy transmission grid

From the scheme of Figure 32(a), the new logical scheme of Figure 32(b) can be derived.

The logical schemes of the components N_G, N_L, N_S and A_L (not shown, for brevity) are obtained by grouping the main electrical equipments (transformer, bus-bar, breaker, switch, power line and protection) following an approach that has the advantage to simplify the logical representation.

The component N_S represents the parts common to all substations (e.g., the bus-bar). Breakers, switches, transformers and protection logics, which are physically part of a substation, are now included in the scheme for the new logical component A_L. In this way, only a few types of different A_L have to be considered. N_G and N_L represent a generation plant and a load, respectively.

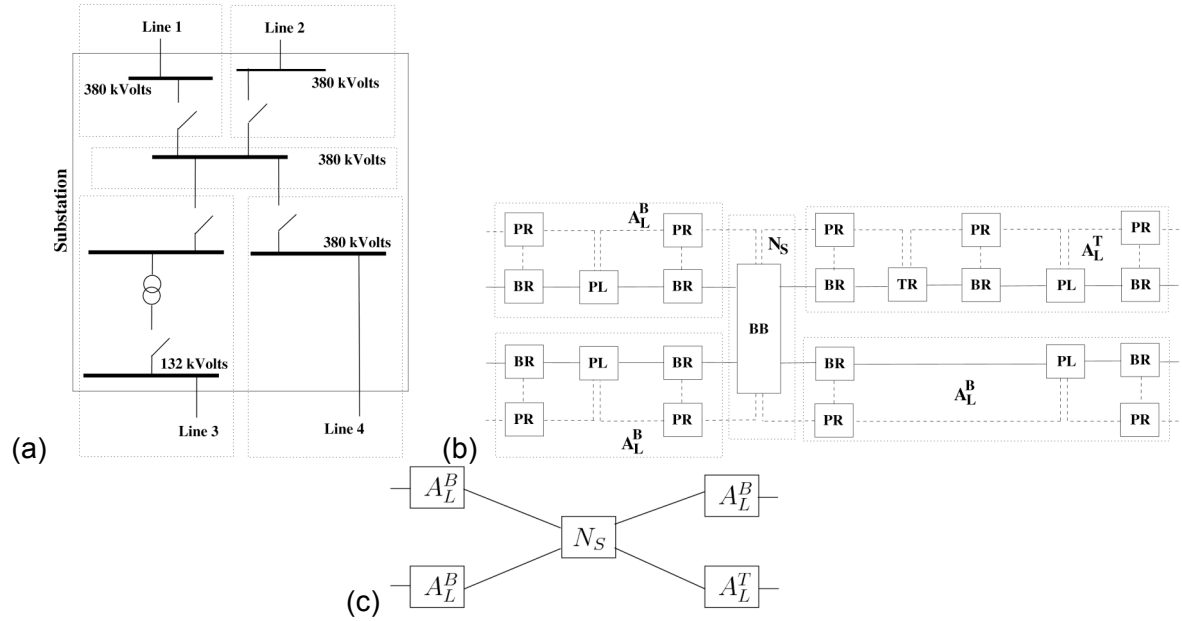


Figure 33: Example of scheme for a substation and the connected power lines: physical scheme (a), low level (b) and high level (c) logical schemes

In Figure 33 an example of physical and logical scheme for a substation and the connected power lines is shown, where two different types of component A_L are considered: $A^{B,L}$ and $A^{T,L}$.

4.2.2 The Information Infrastructure (*II*)

II implements the control system based on information technology. As shown in Figure 34, The main logical components of the *II* are:

- the protection system (*PS*),
- the frequency regulation system (*FRS*), which consists of a primary frequency system at the local level and of a secondary frequency regulation at the upper level in the hierarchy
- the voltage regulation system (*VRS*),
- the teleoperation (or telecontrol) system of the transmission grid (*TTOS*),

- the teleoperation (or telecontrol) system of the distribution grid (*DTOS*),
- the *TSO* transmission network (TSOcommNetw) and the *DSO* transmission network (DSOcommNetw).

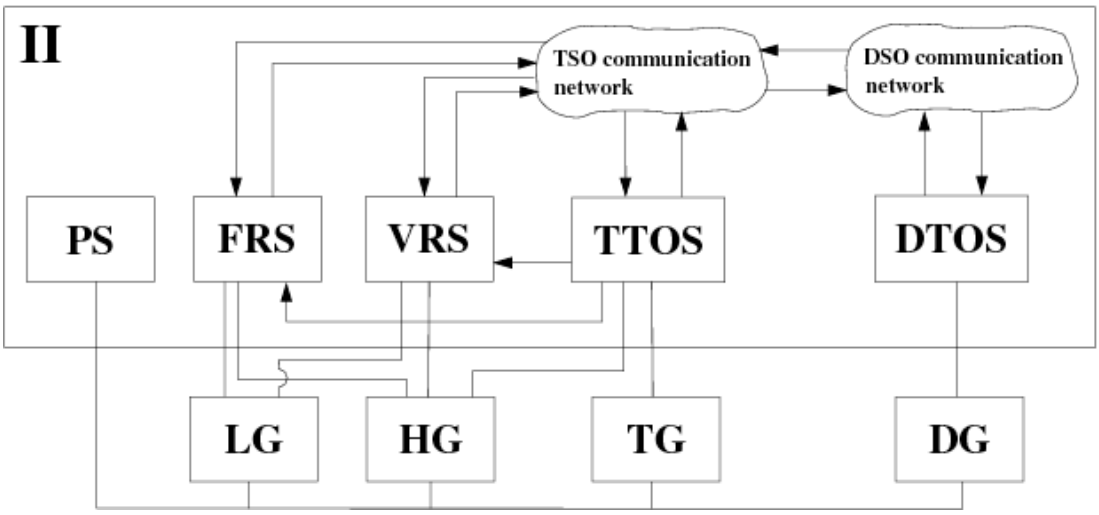


Figure 34: Logical scheme of the II

The protection system is composed of a set of independent (or loosely connected) local protections. We can consider one local protection for each breaker of the EI. FRS has the goal to regulate the frequency of the single generators and along the transmission. It can receive information on the state of the grid from TTOS. VRS has the goal to guarantee that values of the voltages remain as constant as possible along the transmission grid in order to supply to the customers a voltage with good quality without interruptions.

At the current stage, the detailed logical structure of these components and that of the other subsystems involved in the II system are not addressed, and the following discussion will only be limited to the TTOS and DTOS subsystems. In Figure 35 we depict a possible logical structure of TTOS and DTOS, where the components *LCS*, *RTS* and *NTS* of *TTOS*, and the components *LCC* and *ATC* of *DTOS* differ for their criticality and for the locality of their decisions.

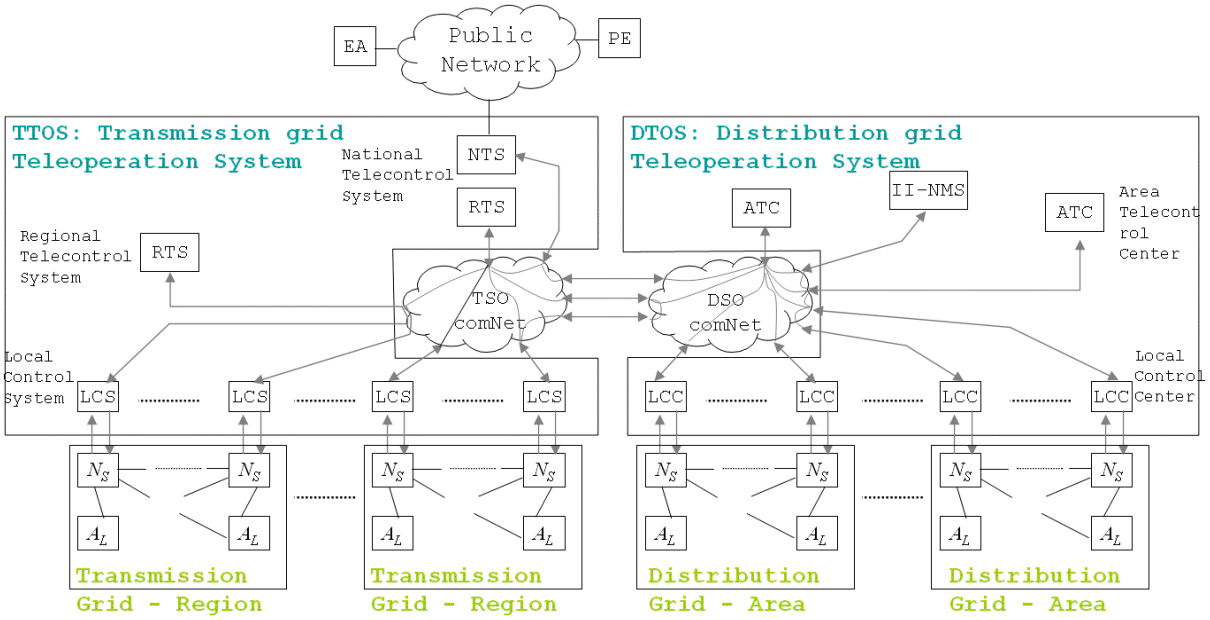


Figure 35: Logical scheme of TTOS and DTOS

The transmission and distribution grids are divided in homogeneous regions and areas, respectively. LCS (Local Control System) and LCC (Local Control Center) guarantee the correct operation of substation equipment and reconfigures the substation in case of breakdown of some apparatus. They include the acquisition and control equipment (sensors and actuators). RTS (Regional Telecontrol System) and ATC (Area Telecontrol Center) monitors their region and area, respectively, in order to diagnose faults on the power lines. In case of breakdowns, they choose the more suitable corrective actions to restore the functionality of the grid. Since the RTS and ATC were not directly connected to the substations, the corrective actions to adopt are communicated to the LCS or LCC of reference. NTS has the main function of supervising the entire grid and handling the planning of medium and long-term operations. NTS also assists the RTS (and ATC) to localize breakdowns on the power lines situated between two regions (two areas). LCS and LCC, such as RTS and ATC, cooperate to decide operation of load shedding.

4.3 State definition for EI and II

The state of the Electrical Infrastructure (EI) is an hybrid-state composed by a discrete part and a continuous one. It can be defined as a 7-tuple (*T, V, F, I, A, P, Q*), where:

- T represents the topology of the grid, i.e., the components N_S, N_G, N_L and A_L and their connections (as shown for example in Figure 32(b)). T could also include information on the direction of the current flow on each power line.

  This information is used to reconfigure the topology of the grid. Therefore, T can be described as an oriented graph where N_S, N_G and N_L are nodes and A_L are arcs.

- V, F, I, A, P and Q are the voltage, the frequency, the current flow, the angle, the active and reactive power associated to the components N_S, N_G, N_L and A_L (if applicable).

T represents the discrete part of the EI states, whereas V, F, I, A, P and Q represent the continuous part of the EI states [CESI RICERCA 2006a].

Following the state model presented in [CESI RICERCA 2006a], each state NORMAL, ALERT, EMERGENCY, IN EXTREMIS and RESTORATIVE of the EI can be described with different combinations of values of the 7-upla (*T, V, F, I, A, P, Q*).

For what concerns the II state, we envision that such a state to be discrete, in the sense that it is only composed by discrete values. Possible states are those identified in Section 3.1.1.1 (i-working, partial i-outage, i-weakened).

4.4 Failure model of EPS and Interdependencies

The failure model of the Electrical Power System (EPS) is presented in three steps. First, the failure model of the Electrical Infrastructure (EI) is sketched. Then, the failure model of the Information Infrastructure is discussed. The third step consists of the II-EI failure model, where the reciprocal impact of II failures and of EI failures is analysed. Therefore, the model assumed for the EI failures and for the II failures is based on their effects on the state of the EI.

4.4.1 Failure model of EI

A disruption (or disturbance or contingency) is the unexpected failure or outage of a EI component, such as generator, power line, circuit breaker, bus-bar, or other electrical components. The main (electrical) disruptions, based on their effects on (single or multiple) components N_S, N_G, N_L and A_L, could be summarized in:

1) Transient or permanent disconnection of a component A_L, N_S, N_G, or N_L with the consequent separation of one or more components from the grid.
2) Transient or permanent failed disconnection of a component A_L, N_S, N_G, or N_L without isolation from the grid.
3) Transient or permanent overloads of A_L, N_S, N_G, or N_L.
4) Unexpected reduction of production of N_G.
5) Unexpected increase or reduction of demand of N_L.
6) Voltage collapse.
7) Underfrequency and loss of synchronism.

The disruptions listed at points from 3) to 7) represent changes of the electrical parameters of the components of the grid N_S, N_G, N_L and A_L.

The disruptions listed at points 1) and 7) represent changes of the topology of the grid T.

After the change of T, at least one or a combination of the values for *V, F, I, A, P* and *Q* will change. Whereas, when *V, F, I, A, P* and *Q* change, the topology T does not change.

4.4.2 Failure model of II

The failures of the II components can be summarized in:

- (transient and permanent) omission failure,
- time failure,
- value failure and
- byzantine failure.

Here the focus is on the failures and not on their causes (internal HW/SW faults, malicious attacks, etc.).

4.4.3 II-EI Failure model (interdependencies)

First, the impact of II failures on EI is analysed. Failures in the II impact on the state of the EI (i.e., on the topology T and on the values of V, F, I, A, P and Q), depending on the logical components affected by the failures, and obviously on the type of the failures.

For example, consequences of a failure of the component *LCS* associated to a component N_S, N_G, N_L and A_L can be:

- *Omission failure of LCS, fail silent LCS***.** No (reconfiguration) actions are performed on the components N_S or A_L.
- *Time failure of LCS.* The above (reconfiguration) actions on the components N_S or A_L are performed after a certain delay (or before the instant of time they are required).
- *Value failure of LCS.* Incorrect closing or opening of the power lines A_L directly connected to the component is performed. These events can occur both when the state of EI requires an action from II (which is incorrect), and when the state of EI is normal and no action from II is actually required.

Failures of the component *LCS* can also impact on the input values that the components *RTS* receive from *LCS*. These values can be omitted, delayed (or anticipated) or erroneous. Since reconfigurations required by *RTS* (or *NTS*) are actuated by the associated component *LCS*, a failure of a component *LCS* can also impact on the reconfigurations required by *RTS* (or *NTS*).

The failure of the component *RTS* (or *NTS*) corresponds to an erroneous (request of) reconfiguration of the state of the EI (including an unneeded reconfiguration) affecting one or more components of the controlled region. The effect of the failure of *RTS* (or *NTS*) on a component N is the same as the failure of the component *LCS* associated to the component N. In the case of Byzantine failure these effects can be different for each component N.

In general, the failure of the components *LCS*, *RTS* and *NTS* may depend on the failures of the components connected to them through a (public or corporate) communication network.

Disruptions of EI on II constitute a physical interdependency. Disruptions of the EI infrastructure impact on (parts of) the II system by lessening its functionalities (till complete failure in the extreme case the disruption is a total blackout of the power grid). For example, a disruption may cause a partial blackout, that reduces the performance of the private or public network used by the II. Then, the communication times degrade, leading to timing failures of the II.

4.5 Dynamic behaviour of EPS

The hybrid-state of EI changes when the topology T of the system or the values for *V, F, I, A, P* and *Q* change, i.e., when one of the following events occurs:

- disruption (including failure of a local protection),
- activation of a protection local to the EI,
- voltage or frequency regulation or reconfiguration action by II (including erroneous, delayed or not required action),

- maintenance actions on the EI.

Therefore, the state of EI can also change due to actions by the II. These actions can be correctly activated by an event in the EI, or can be erroneously activated by a failure of the II.

The discrete-state of II can change when one of the following events occurs:

- failure of a component of the II,
- disruption of the EI,
- recovery.

To better describe the interaction between EI and II, we show in Figure 36 and Figure 37 an example of a possible temporal evolution of the EI system after a fault which breaks a line, in absence and in presence of II, respectively.

As in [CESI RICERCA 2006a], let us denote with NORMAL, ALERT, EMERGENCY and IN EXTREMIS the set of operative states of EI, where the criticality of the system increases from NORMAL (situation in which all the constraints are satisfied) to IN EXTREMIS (in which the service is partially or totally interrupted). Each of these states can be described with different combinations of values of the 7-tuple (T, V, F, I, A, P, Q). At time 0, EI is in a state NORMAL $S_0=(T_0, V_0, F_0, I_0, A_0, P_0, Q_0)$. At time t_F, a disruption, due to a tree fall, causes a loss of a line and EI moves to the degraded state ALERT $S_F=(T_F, V_F, F_F, I_F, A_F, P_F, Q_F)$.

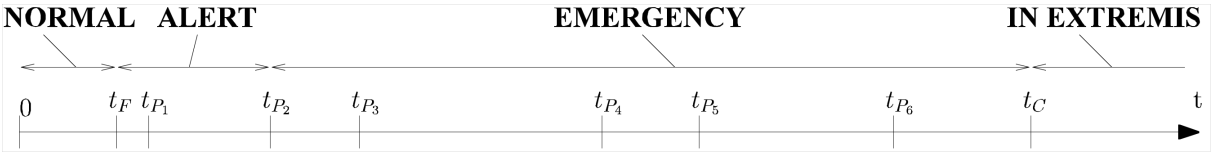


Figure 36: EI behaviour in absence of II

In Figure 36, at the instants of time t_{P1}, t_{P2}, …, t_{P6} six activations of protections isolate components of EI, and EI moves into new degraded states until reaching, at time t_C, the state IN EXTREMIS, where the total service interruption cannot be avoided.

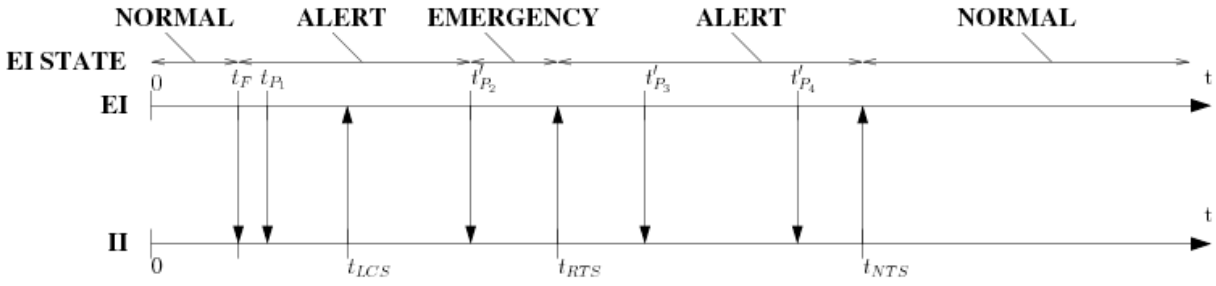


Figure 37: EI behaviour in presence of II

In Figure 37, II is also considered. Then, at time t_F three types of activities of the components LCS, RTS and NTS start on II. When these activities complete at the instants of time t_{LCS}, t_{RTS} and t_{NTS}, EI should move into a state less degraded than the state in which EI would be without considering II.

4.6 Measures of interest for the EPS

Dependability analysis of EPS based on a stochastic approach requires the definition of measures of performability, which is a unified measure proposed to simultaneously deal with performance and dependability.

A set of measures specific for the EPS can be based on the following reward structure where costs and rewards are considered with respect to the point of the view of the power producers and distributors:

- To each generator a cost is associated, depending on: the generated power P, the type of generator, the economic loss implied by a breakdown of the generator.
- To each load a positive reward is associated, depending on: the consumed power, the criticality of the load.
- To each interruption of service supply a cost is associated, depending on: the difference between the required power and the available power for each load, the number of loads which will be powered off, the criticality of loads which will be powered off, duration of the interruption.

Using this reward structure, we can evaluate the expected reward at time t and in the interval [0, t], defined by the cost (or reward) associated to each generator i minus the reward (or cost) associated to each load j.

Other possible measures of interest are the following:

- The expected percentage of delivered power (the delivered power divided by the power demand) at time t and in the interval [0, t];
- The expected numbers of components affected by a disruption at time t and in the interval [0, t].

4.7 Prominent Aspects of the EPS modelling framework

To model and evaluate the performability of EPS we first define a model representing the behaviour of the system at the needed level of detail, and then we solve it by simulation or analytically.

To represent and model the behaviour of EI and II and their interactions, the following aspects should be considered.

Structural aspects:

- The system has a natural hierarchical structure, as shown in the examples of logical schemes of Section 4.2.
- At a certain level of detail, the system is composed by many similar components having the same logical structure, as shown, for example, in Figure 32(b) for the logical components N_S, N_G, N_L and A_L. In effect, these components can be grouped based on similar sub-components. All similar components can be considered as non anonymous replicas having the same structure and different parameter values for the activities and the events represented.
- The topology of the grid and the electrical values associated to each component of the grid are part of the state of the EI.

Behavioural aspects:

- The time to disruptions of the components N_S, N_G, N_L and A_L depends also on the value of the electrical parameters associated to the components. A disruption of a component can propagate to contiguous components.
- The propagation time of a disruption should not be considered instantaneous.
- Protections can stop the propagation of a disruption by isolating from the grid the component affected by a disruption. The activation time of a protection should not be considered instantaneous. The correct activation of a protection depends also on the strength of the disruption and on the value of the electrical parameters associated to the protection component.
- The reaction time (with respect to the occurrence of a disruption), the failure time and erroneous activation time (when no disruptions have occurred) of a component (e.g., *LCS*, *RTS* and *NTS*) should be considered.
- The functions that implement the reconfiguration and regulation algorithms should be considered. These functions activate when EI is not in equilibrium; in such conditions, they receive as input the 7-tupla (T_i, V_i, F_i, C_i, A_i, P_i, Q_i) where the EI is not in equilibrium and produce as outputs the new 7-tupla (T_e, V_e, F_e, C_e, A_e, P_e, Q_e) which allows the equilibrium condition to be restored (that is, EI is back to the NORMAL state), if possible.

To capture the above discussed structural and behavioural aspects, the modelling and evaluation framework should possess the following major characteristics, grouped into three categories: modelling power aspects (the basic modelling mechanisms required to build the EPS model), the modelling efficiency aspects (the advanced modelling mechanisms required to build the EPS model more efficiently), and the solution power aspects.

Modelling power:

[CharA1] Different formalisms for different sub-models.

[CharA2] Representation of continuous, discrete and hybrid state.

[CharA3] Time distributions, probability distributions and conditions enabling the time consuming events which depend on the discrete or continuous state.

[CharA4] The call to the function which implements the reconfiguration and regulation algorithms.

[CharA5] Definition of performability measures.

Modelling efficiency:

[CharB1] Hierarchical composition of the model.

[CharB2] Replication of (anonymous and non anonymous) sub-models.

[CharB3] Compact representation for the topology of the grid (for T), for example, describing a part of the state of the system in terms of a matrix (incidence matrix [nodes x arcs]).

[CharB4] Compact representation of continuous state (for *V, F, I, A, P* and *Q*), for example, describing a part of the state of the system in term of arrays, associating to each component of the EI grid (nodes and arcs) the values of *V*, *F, I, A, P* and *Q* (if any).

Solution power:

[CharC1] Analytical solution of the overall model (if possible). Problems could be: the explosion of the states of the model; an analytical solution method could not

exist for the class of model considered, depending on the considered time distributions; the stiffness.

[CharC2] Simulation (by existing automatic tools or ad hoc simulation software).

[CharC3] Separate evaluation of different sub-models and combination of the results.

4.8 On the construction of the overall EPS model

In this Section we address the problem of building the overall model for the entire EPS, considering the logical scheme of the electrical grid as shown in Figure 32(b).

The model construction should consist of the following steps:

1. Define the models M_N and M_A for each generic component N (representing a node of the grid, with N= N_S, N_G, N_L) and for A=A_L (representing an arc of the grid). To simplify the example we do not consider different schemes for each component.
2. Duplicate M_N and M_A for each specific component N and A, and set its individual parameters.
3. On the basis of the topology T, connect manually, the models M_N and M_A, by using a composition operator of the models M_N and M_A, for each node N connected to an arc A.

When the number of components N and A is high, the construction of the model based on the above approach can be very expensive in terms of time and very error prone. The above process could be automated, defining a function which receives in input an incidence matrix describing the topology T of the grid and generates the composed model representing T.

Alternatively, a model describing a topology can be defined by using a compact approach based on replication and possibility to define part of the state of a system with an array (for the incidence matrix [nodes x arcs]). In this case, to construct a model representing a topology like that shown in Figure 32(b), the following steps should be required, for m nodes N and n arcs A:

1. Define the model M_N and M_A for each generic component N (representing a node of the grid, with N= N_S, N_G, N_L) and for A=A_L.
2. Define a part of the state of M_N and M_A by using a matrix (m x n) T[i,j] of binary values (0,1), where T[i,j]=1 if the component *i-th* is connected to the component *j-th*, otherwise T[i,j]=0 (the values 1 and -1 can be used if it is needed to represent also the direction of the arc, i.e., if T is an oriented graph). The time distributions and the conditions in the model M_N can depend on the values of T. In particular, the *i-th* replica of M_N (or the *j-th* replica of M_A) can be defined as a function of T[i,j], and can modify T[i,j] (see below).
3. Define a hierarchical model by automatically replicating m times the model M_N, by assigning to each replica a different index, from *1* to m. The state defined with matrix T is common to all the replicated sub-models M_N. The parameters of the *i-th* replica can depend on the values of T[i,j].
4. Define a hierarchical model by automatically replicating n times the model M_A, by assigning at each replica a different index, from *1* to n. The state defined with matrix T is common to all the replicated sub-models M_A. The *j-th* replica can depend by the values of the element T[i,j].

Thus, it is possible to model a sub-system without constructing (or duplicating) manually the models for each single component *N* and *A* of the grid and without connecting each specific couple of models manually to obtain the required topology.

Following the same compact approach it is also possible to define the parameters of the replicas of the model M_N (or M_A) as a function of continuous state and to model fault propagation.

4.9 High-level description of the EPS's behaviour

In this section we provide a high-level description of the behaviour of some of the main EPS components using flowcharts diagrams. The goal is to give the reader an overall view of what has been modelled without detailing how it has been modelled, focusing on two main elements of the II infrastructure, *RTS* (Section 4.9.1) and *LCS* (Section 4.9.2), on the EI autoevolution function (Section 4.9.3), and on a set of basic EI components: Nodes (Section 4.9.4), Power lines (Section 4.9.5) and Breakers (including protections, Section 4.9.6).

4.9.1 RTS behavior

In Figure 38 it is depicted the flowchart representing the Regional Telecontrol System (*RTS*) behaviour, focusing on its interdependencies with the EI. In particular, we aim to describe how the *RTS* reacts when a change of the EI state is detected. Here we are completely abstracting from the causes that trigger the EI state change, thus focusing on the effects that a state change in EI cause on the state of II.

Three variables have been defined:

- *RTS* (string), which is equal to 'NULL' if the Regional Telecontrol System is not currently activated to perform a reconfiguration action, otherwise it is equal to 'BEGIN';
- *TimeToRTS* (integer), which represents the remaining time needed by *RTS* to execute the reconfiguration action (that is supposed to be applied instantaneously);
- *STATUS RTS* (string), which is equal to 'TIME_FAILURE', 'VALUE_FAILURE' or 'NULL' if the Regional Telecontrol System is currently affected by a time failure, by a value failure or if it is not affected by any failure, respectively.

For better describing the dynamics of the two infrastructures, the system's lifetime is seen as a sequence of time units (time discretization).

In this scenario we are assuming that each time the state of EI changes, the computation of the *RTS* reconfiguration action starts from the beginning (the variable *TimeToRTS* is set again to its initial value), since it is not working on the most updated EI state.

Once *RTS* completes the computation, it can trigger and instantaneously apply the correct or incorrect reconfiguration action depending on its state (it depends on the value of *STATUS RTS*). The application of the correct or incorrect *RTS* reconfiguration action can also be delayed if in the meantime *RTS* has been affected by a timing failure (*STATUS RTS* == *TIME_FAILURE*). A correct *RTS* reconfiguration action will find a new global (so optimal) equilibrium considering dispatching/shedding operations.

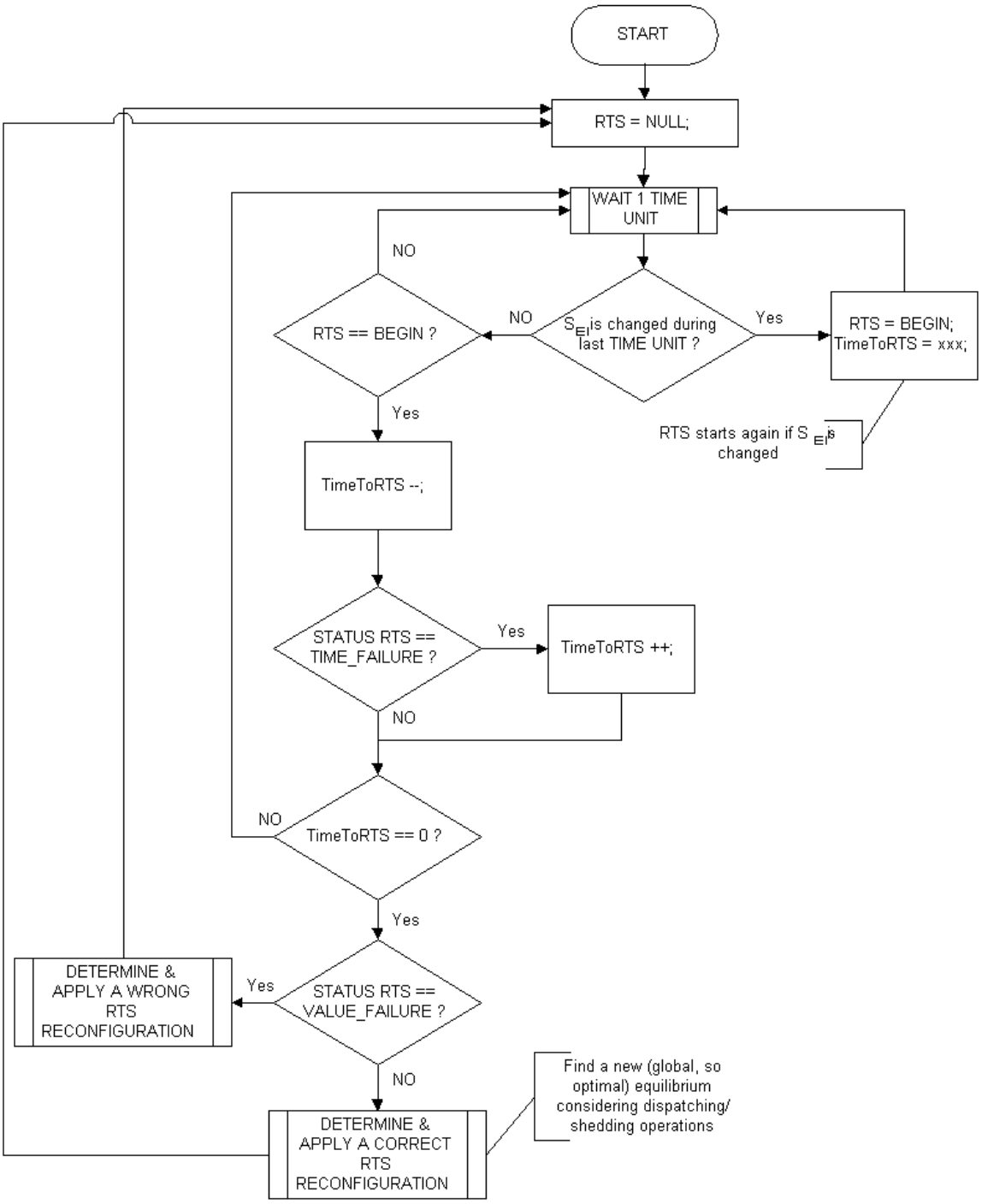


Figure 38: RTS behaviour

4.9.2 *LCS* behaviour

Figure 39 shows the behaviour of the Local Control System associated to each substation. The variable *LCS STATUS* represents the status of *LCS*, that can be failed or not. The reconfiguration action cannot be applied by *LCS* if it is failed, and in this case it does not perform any reconfiguration (fail-stop behaviour) and it is isolated to be repaired. The *LCS* reconfiguration actions can be triggered by *RTS* (global reconfigurations) or can be directly triggered by *LCS* when it locally detects that the (electrical) equilibrium has been violated..

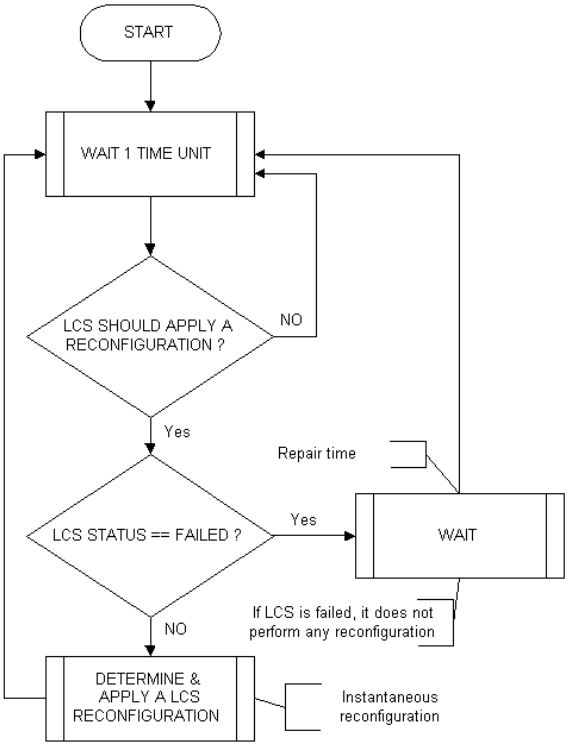


Figure 39: LCS behaviour

4.9.3 EI autoevolution

Figure 40 describes the automatic evolution of EI when an event modifying its state occurs. When S_{EI} changes, EI tries to find a new equilibrium for the new grid topology, letting the generated and consumed power unchanged (only redirections of current flows). The new equilibrium is reached instantaneously and no II actions are performed.

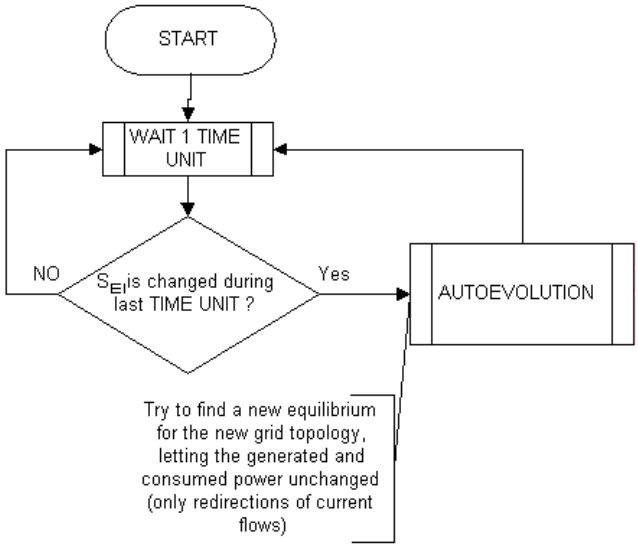


Figure 40: EI autoevolution

4.9.4 Node (generator/substation/load) behaviour

Figure 41 depicts the behaviour of a node of EI, which could be a generator, a substation or a load. The variable *STATUS NODE* represents the status of node that can be correctly working (*STATUS NODE == OK*) or failed (*STATUS NODE == DISRUPTED*). When the node is failed it is put under repair.

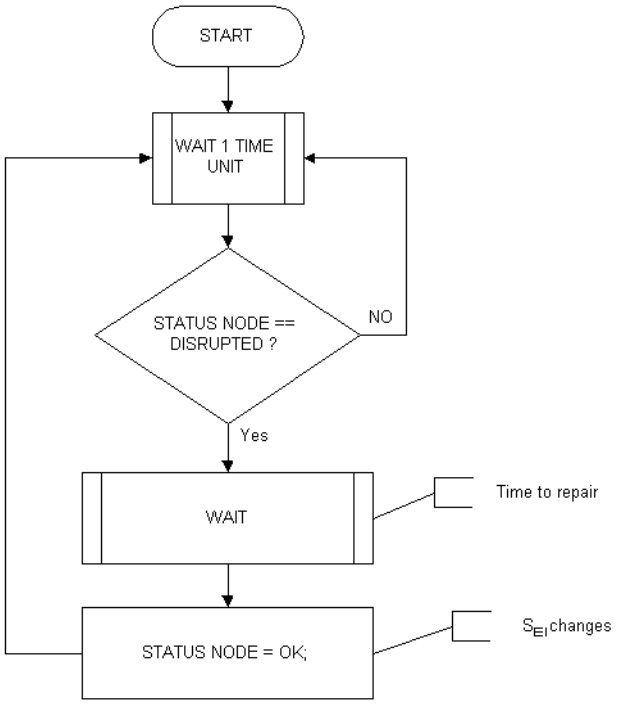


Figure 41: Node behaviour

4.9.5 Power line behaviour

Figure 42 depicts the behaviour of a power line. Variable *STATUS PL* represents the status of the power line that can be correctly working (*STATUS PL == OK*) or failed (*STATUS PL == DISRUPTED*). When the power line is failed it is put under repair.

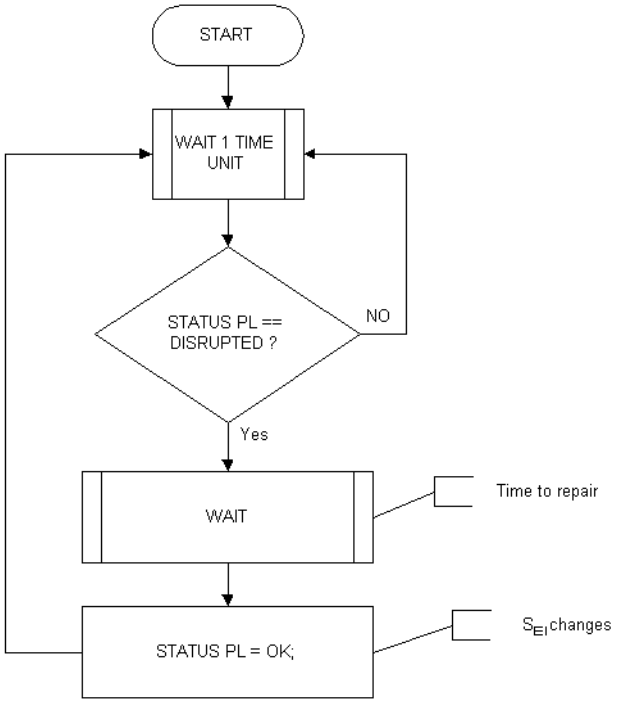


Figure 42: Powerline behavior

4.9.6 Breaker behaviour

Figure 43 represents the breaker behaviour. We indicate with breaker both the breaker and the associate protection that triggers its reconfigurations.

The variable *STATUS BREAKER* represents the status of the breaker: it can be correctly working and open (*STATUS BREAKER == OPEN*), correctly working and closed (*STATUS BREAKER == CLOSE*), failed as stuck open (it cannot be closed, *STATUS NODE == STUCK OPEN*), and failed as stuck closed (it cannot be opened, *STATUS NODE == STUCK CLOSE*).

If the breaker is open (or stuck open) it tries to close (*CLOSE BREAKER* command). If it is stuck open it needs to be repaired (*REPAIR BREAKER* command). If it cannot be maintained closed (the failures that leaded to its open have not been removed), it tries to close again after a while.

If the breaker is closed (or stuck closed), it should be opened if the electric conditions of the controlled power line and node are anomalous and they can induce a disruption. In this case, the associated breakers should be immediately opened (*OPEN BREAKER* command). If the breaker is stuck closed it cannot be opened and then the disruption (probabilistically) propagates to the controlled power line and node, and the breaker is repaired.

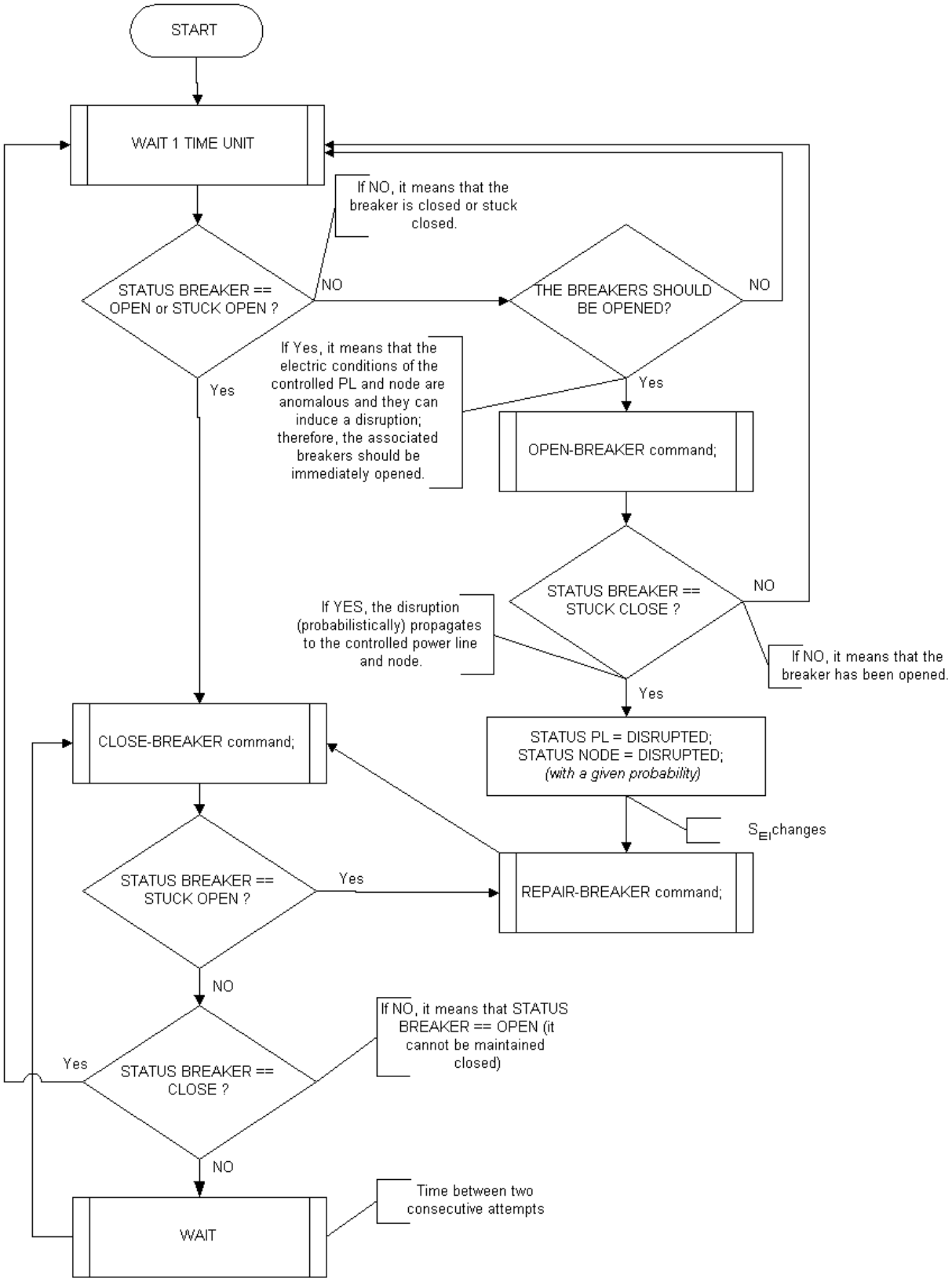


Figure 43: Breaker behaviour

The *OPEN BREAKER* command, the *CLOSE BREAKER* command and the *REPAIR BREAKER* command are described in the following Sections 4.9.6.1, 4.9.6.2 and 4.9.6.3, respectively.

4.9.6.1 *OPEN BREAKER* command

Figure 44 describes the *OPEN BREAKER* command introduced in Figure 43. If the breaker is already opened (*STATUS BREAKER == OPEN* or *STUCK OPEN*), then the command has no effect. If it is stuck closed, it goes under repair, while if it is closed it opens (*STATUS BREAKER = OPEN*).

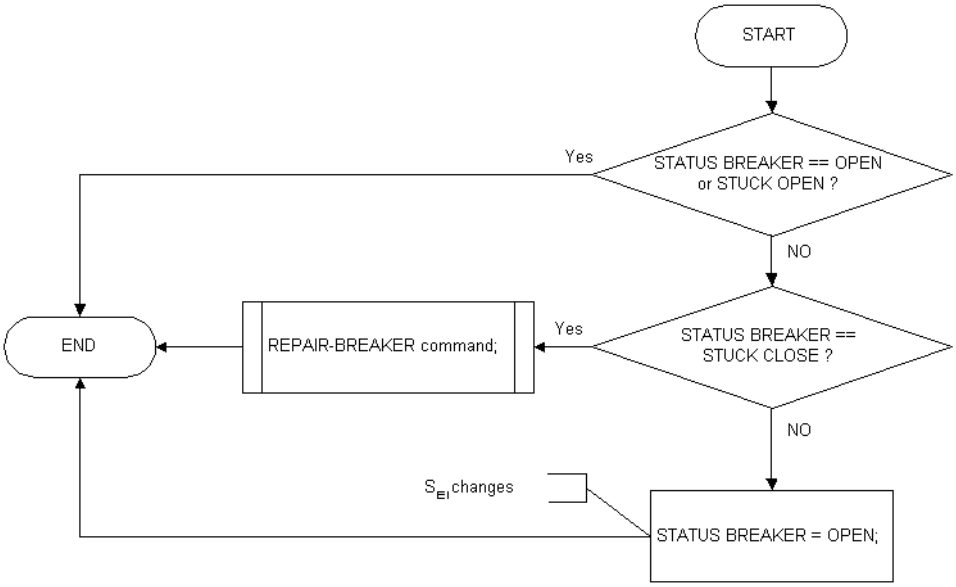


Figure 44: OPEN BREAKER command

4.9.6.2 *CLOSE BREAKER* command

Figure 45 describes the *CLOSE BREAKER* command introduced in Figure 43. If the breaker is already closed (*STATUS BREAKER == CLOSE* or *STUCK CLOSE*), then the command has no effect. If it is stuck open, it goes under repair. The breaker actually closes if it is open and if it can be maintained closed. A breaker can be maintained closed if the failures that leaded to its open have been removed, otherwise the breaker remains open.

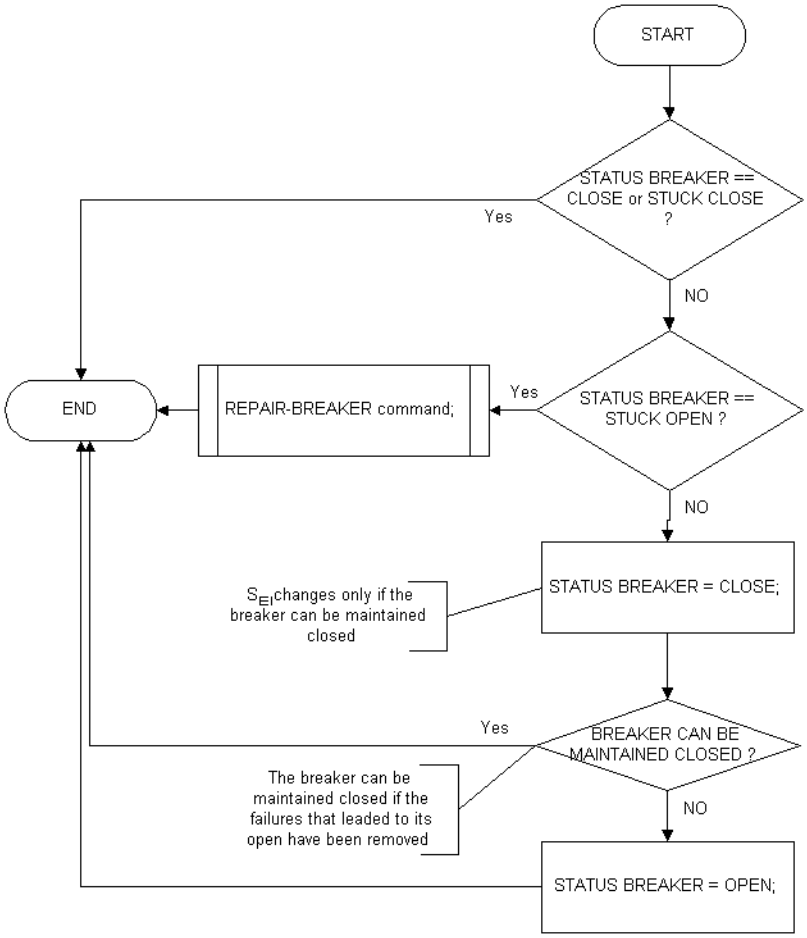


Figure 45: *CLOSE BREAKER* command

4.9.6.3 *REPAIR BREAKER* command

Figure 46 describes the *REPAIR BREAKER* command introduced in Figure 43. If the breaker is stuck open then it must be repaired. If the breaker is stuck close, before repair the power line needs to be opened, and this procedure may take some time.

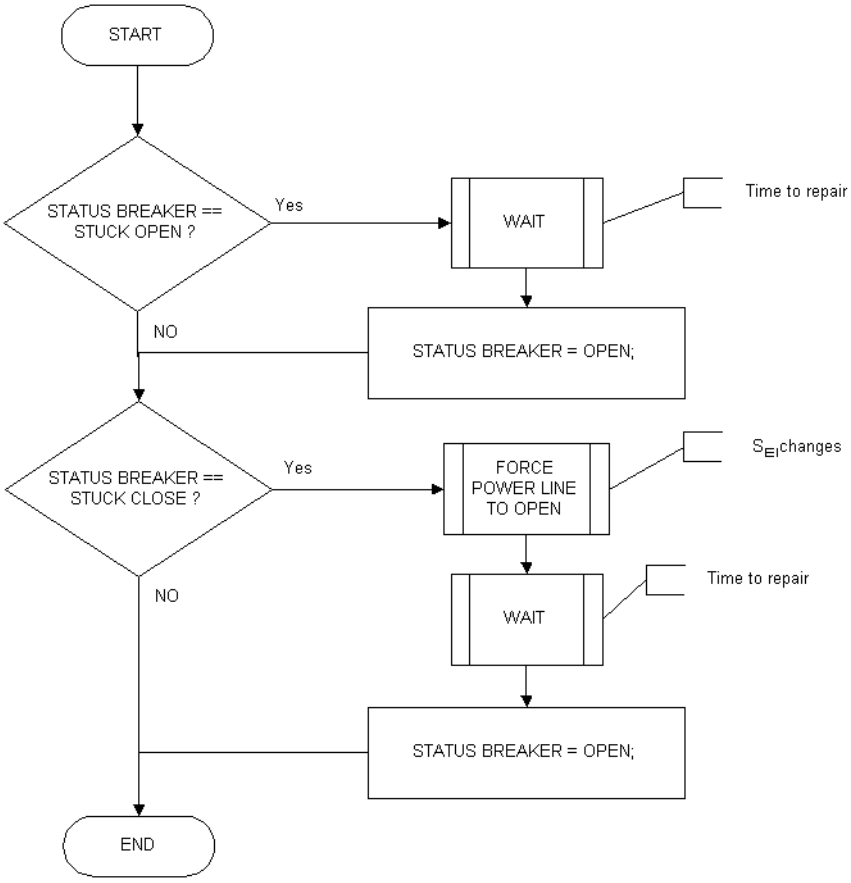


Figure 46: *REPAIR BREAKER* command

4.10 Feasibility of the modelling framework using SAN and Möbius

In this Section we discuss the feasibility of the proposed framework using Möbius [Clark *et al.* 2001], a powerful multi-formalism/multi-solution tool, and presenting the implementation of a few basic modelling mechanisms adopting the Stochastic Activity Network (SAN) formalism [Sanders & Meyer 2001], that is a generalization of Stochastic Petri Nets formalism. Here the goal is not to provide a complete and detailed model representing a concrete instance of an EPS, but to show how some basic framework's characteristics can be actually obtained. To this purpose we describe the model construction of a simple instance of the EPS, focusing on substations (Section 4.10.1), protections (Section 4.10.2), Local Control Systems (Section 4.10.3), Regional Tele-control Systems (Section 4.10.4), and on the overall EPS as composition of different submodels (Section 4.10.5).

The number of components N_G, N_S, N_L and A_L of EI is *n_G*, *n_S*, *n_L* and *n_A* respectively. These components are represented by replicated SAN with index i, with:

- i ∈ [*0,n_G-1*] for N_G,
- i ∈ [*n_G,n_G+n_S-1*] for N_S,
- i ∈ [*n_G+n_S,n_G+n_S+n_L-1*] for N_L,
- i ∈ [*n_G+n_S+n_L,n_G+n_S+n_L+n_A-1*] for A_L.

4.10.1 Modelling a substation

In Figure 47 the SAN for the component N_S is shown.

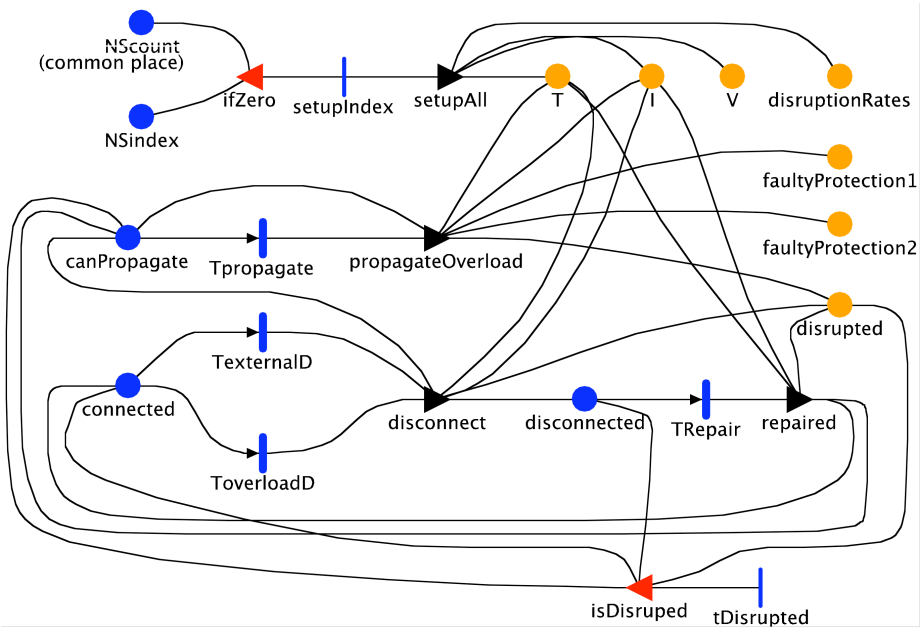


Figure 47: SAN for N_S component

In describing the SAN elements, we make explicit reference to the framework's characteristics, among those identified in Section 4.7, here implemented as basic modelling mechanisms. The place *NSindex* represents the index of the component N_S. The place *NScount* is common to the replicas of the SAN and contains *n_S* tokens at time *0*. The extended place T (common place among the replicas) is an array of array of $n\_A \cdot n\_N$ short type, with *n_N=n_G+n_S+n_L*, and represents the topology of the transmission grid and its state (CharA2, CharB3 of Section 4.7). The following value can be associated to each element T–>Index(i)–>Index(j)–>Mark() of T:

-1: for a connection from the arc i to the node j (the current flow enters into the node);

+1: for a connection from the node i to the arc j (the current flow exits from the node);

-2, +2: if the connection has been opened and can be closed by reconfiguration;

-3, +3: if the connection has been opened due to a disruption and can be only closed after a repair

For the sake of simplicity, in this on-going example we only consider two different types of electrical parameters, the current flow *I* and the voltage *V*. The extended places *I* and *V* (common places among the replicas) are arrays of *n_EPS* struct type, with *n_EPS=n_G+n_S+n_L+n_A*, and represent the electrical parameters and the current values associated to each component for the current flow and for the voltage, respectively (CharA2, CharB4 of Section 4.7). For example, for each generator plant *i*, *I–>Index(i)* represents the produced current flow and the maximum current flow which can be produced; for each station *i*, *I–>Index(i)* represents the current flow associated to it, the threshold current flow for an overload, and the threshold current flow for a breakdown of the component. The extended place *disruptionRates* (common place) is an array of *n_EPS* struct type and represents the rate of occurrence of different type of disruptions associated to each component. These rates depends on the component (e.g., the length of the line, the position of the station, etc.) and on the type of disruption (lightning, tree fall, etc.). The extended place disrupted (common place) is an array of *n_EPS* short type and represents the state of disruption (breakdown) of each component. The extended places *faultyProtection1* and *faultyProtection2* (common places) are arrays of *n_A* short type and represent the state of failure of the protections associated to each line.

The activity *setupAll* is enabled when the marking of *NSindex* is equal to *0*, i.e., for each replica of the SAN for which the *NSindex* has not yet been set. The function of the input gate *ifZero* removes one token from *NScount* and set the marking of *NSindex* to *n_G+(n_S-NScount –>Mark())-1*, where *A–>Mark()* is the current marking of *A*. This part of the model enables to distinguish each replica when the N_S model is replicated *n_S* times to build the complete EI infrastructure (CharB2 of Section 4.7). The gate *setupAll* is executed only by the first replica, and it sets all the parameters and the initial state of EI represented by the extended places by executing the following C++ like code:

```
if( NSindex->Mark()==nG)  {
   setupT(configTfile, T, nA, nN);
   setupI(configIfile, I, nEPS);
   setupV(configVfile, V, nEPS);
   setupDisruptionRates(configDisRatefile, disruptionRates, nEPS); }
```

The functions *setupT()*, *setupI()*, *setupV()* and *setupDisruptionRates()* set the initial values of the extended places by reading the configuration values from the input files. For example, the input file *configTfile* can have a row for each electric line with the following format: "*nodeIndex arcIndex nodeIndex*". The activity *Tpropagate* represents the occurrence of an event of overload of current flow (e.g., *lightnings*) which can propagate instantaneously to the other components. This activity has exponential distribution with rate

disruptionRates–>Index(NSindex–>Mark())–>overloadPropagation–>Mark().

When this activity completes the code of the gate *propagateOverload* is executed:

```
propagateOverload(T, disruped, faultyProtection1, faultyProtection2, T, NSindex->Mark());
autoevolution(T, I, V);
if( disruped->Index(NSindex->Mark())->Mark()==0 ) canPropagate->Mark()=1;
```

The function *propagateOverload()* receives in input the topology state T, the disruption state of the components disrupted, the state of the failure of the protections *faultyProtection1* and *faultyProtection2*, the current flow on the grid and the index of the component affected by the overload which must propagate (CharA4 of Section 4.7). The result of the execution of *propagateOverload()* is that the values of T, disrupted and *faultyProtection1* and *faultyProtection2* are modified, due to the propagation of the overload. The function *autoevolution()* updates *I* and *V* for the new topology T. This function represent the automatic evolution of the values of *I* and *V* when the topology changes. It must be executed after each change of T.

The activities *TexternalD* and *ToverloadD*, which represent the time to the occurrence of an external (e.g., a tree fall) or internal (e.g., a disruption due to the current flow) disruption, are exponential. The rate of *TexternalD* is

disruptionRates–>Index(NSindex–>Mark())–>externalDisconnection–>Mark().

The rate of *ToverloadD* depends on the current flow of the component (CharA3 of Section 4.7), and can be obtained by the following C++ like code:

```
if(I->Index(NSindex->Mark()) ->curr->Mark() <
        I->Index(NSindex->Mark())->overload->Mark())
   return(disruptionRates-> Index(NSindex->Mark())->agingRate->Mark() );
else
   return( overloadedDisruptionRate(I, disruptionRates->Index(NSindex->Mark())
```

-> agingRate-> Mark(), NSindex->Mark())}

The function *overloadedDisruptionRate()* returns the rate of disruption when the current flow is greater than a threshold *I–>Index(NSindex–>Mark())–>overload–>Mark()*. The gate disconnect change the topology state T when a component is disrupted. For each index of row i of T for which *T–>Index(i)–>Index(NSindex–>Mark())–>\textgreaterMark()* is equal to 1 or -1 then this value is changed to 3 or -3, respectively, and the following code is executed:

```
disconnected->Mark()=1;
canPropagate->Mark()=0;
disruped->Index(NSindex->Mark())->Mark()=1;
autoevolution(T, I,V);
```

After the repair of the component, represented by the activity *TRepair*, the values of T and disrupted are updated by the output gate repaired and the function *autoevolution()* is executed.

The immediate activity *tDisrupted* is enabled when, after a disruption propagation, the component is affected by a disruption, i.e., *disrupted–>Index(NSinde–>Mark()–>Mark()==1*) and the local places *canPropagate*, connected and disconnected must be updated.

4.10.2 Modelling protections

As depicted in Figure 33(b), the logical component A_L consists of a power line (*PL* component), two breakers (*BR* components) that can physically disconnect the power line from the two connected substations, and two protections (*PR* components, one for each breaker) that monitor the basic electrical parameters of the power line and possibly trigger a disconnection action. In Figure 48 is shown the SAN representing the protections between the power line and the connected substations. The upper part of the model is similar to the SAN for N_S (see Section 4.10.1), so here we only describe the lower part of the model.

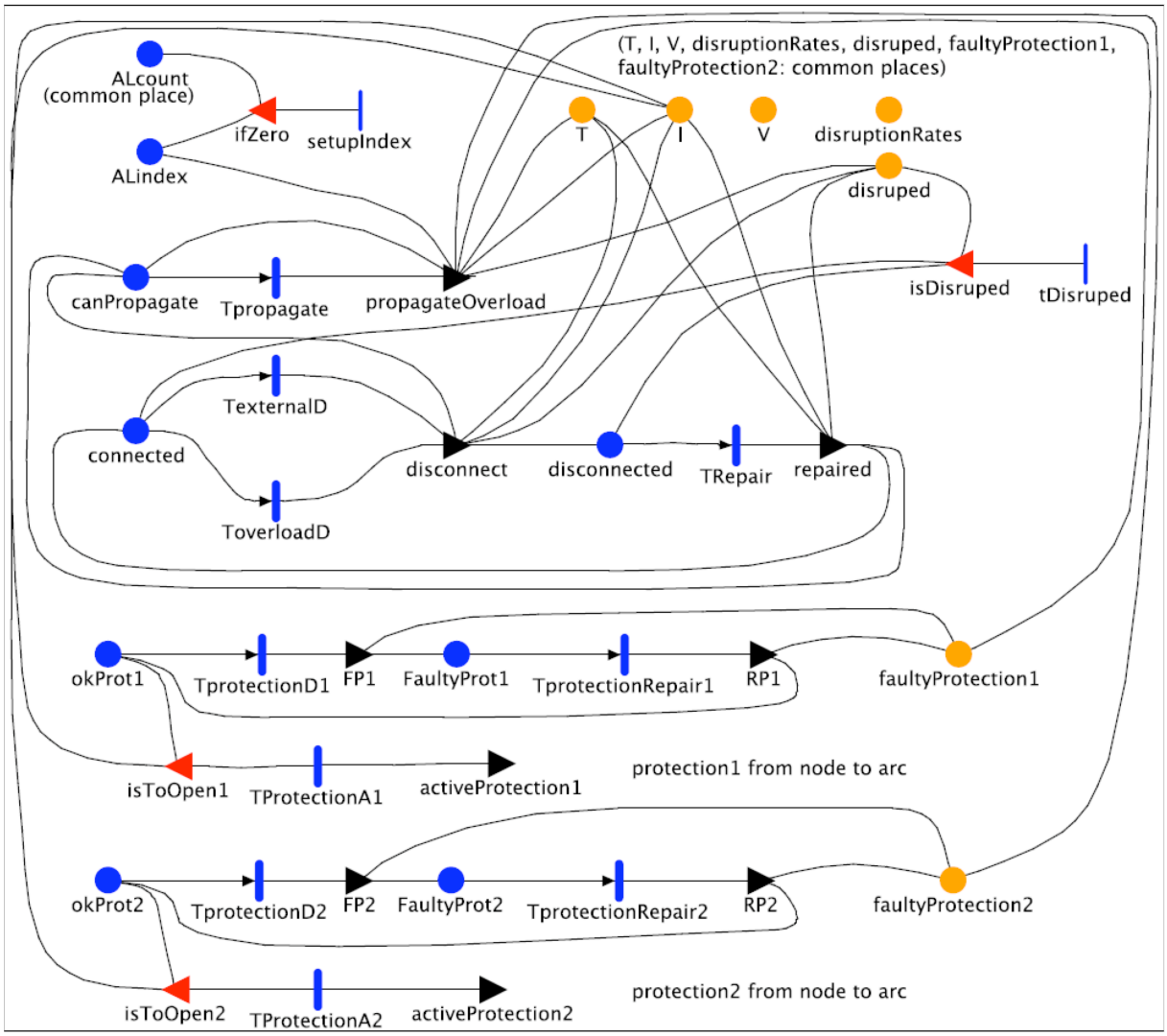


Figure 48: SAN for protections inside A_L component

Let us denote with *Prot1* and *Prot2* the two protections inside A_L. In the following we only describe the SAN concerning *Prot1* since the SAN for *Prot2* is similar. The activities *TprotectionD1*, *TprotectionRepair1* and *TProtectionA1* represent, respectively, the time to the occurrence of a disruption, the time to repair and the time to activation of a protection (CharA3 of Section 4.7). A protection can be activated if it is not failed and if the current flow is grater than a threshold, i.e., the following condition of the gate *isToOpen1* is verified (CharA3 of Section 4.7):

(okProt1->Mark()==1 &&
I->Index(ALindex->Mark())->curr->Mark() >= I->Index(ALindex->Mark())->max->Mark())

The gate *activeProtection1* is executed when a protection is activated and the line is disconnected from the node. In this case, for each index of column *j* of T such that *T–>Index(ALindex–>Mark())–>Index(j)–>Mark()* is equal to 1, this value is changed to 2 (CharB3 of Section 4.7). Moreover, the following code is executed (CharA4 of Section 4.7):

disconnected->Mark()=1;
connected->Mark()=0;
canPropagate->Mark()=0;
autoevolution(T, I, V);

When the protection fails, the value of *faultyProtection1–>Index(ALindex–> Mark()-(n_G+n_S+n_L))–>Mark()* is set to 1 by the gate *FP*1; when the protection is repaired, the gate *RP1* set this value to *0* and the marking of *okProt1* is set to *1*.

4.10.3 Modeling a Local Control System

In Figure 49 is shown the SAN for the component *LCS* (see also Figure 35). The extended place *LTSfailed* (common place) is used for the interactions between the replicas of *LTS* and *RTS*.

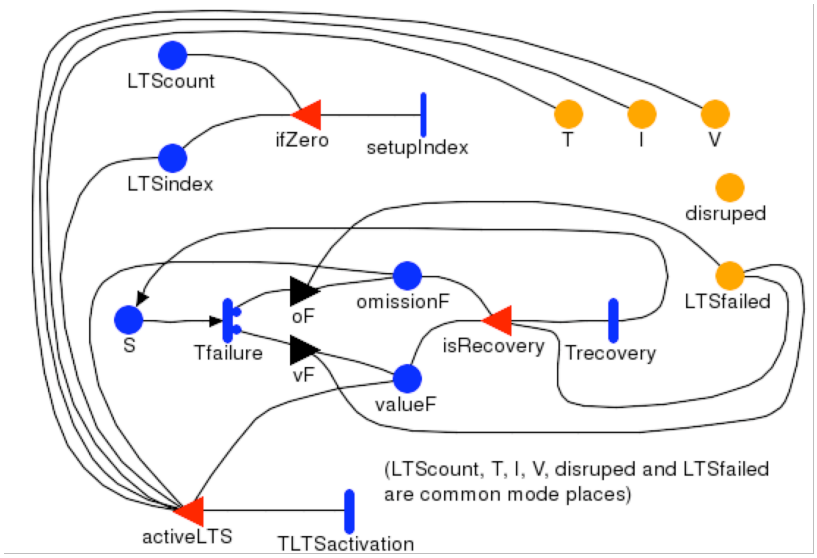


Figure 49: SAN for LCS

The activities *Tfailure* and *Trecovery* represent the time to failure and to recovery of the *LTS*. A failure of *LTS* (that occurs when the activity *Tfailure* completes) may be an omission failure (case1) or a value failure (case2). When an omission failure occurs, the value of *LTSfailed–>Index(LTSindex–>Mark()) –>value–>Mark()* is set to 1 by the gate *oF*. When a value failure occurs, the value of *LTSfailed–>Index(LTSindex–>Mark())–>omission–>Mark()* is set to 1 by the gate *vF*. When a repair occurs (and it happens when the activity *Trecovery* completes), these values are set to 0 by the gate *isRecovery*. The activity *TLTSactivation* represents the time that *LTS* spends to trigger the reconfiguration actions (CharA3 of Section 4.7). This activity is enabled when there is not an omission failure and when the value of the function *LTSenable(T, I, V, LTSindex->Mark())* is true (CharA4 of Section 4.7). The result of the function *LTSenable()* depends on the values of the electrical parameters and on the topology of the controlled components. When the activity completes, the following actions are performed by the gate *activeLTS* (CharA4 of Section 4.7):

LTSreconfigure(T, LTSindex->Mark());
if (valueF->Mark()!=0) LTSchangeValue(T, LTSindex->Mark());
autoevolution(T, I, V);

The function *LTSreconfigure()* can change the part of the topology related to the node with index *LTSindex->Mark()*. The function *LTSchangeValue()* performs the changes of the configuration due to the value failure of LTS, thus representing a potentially wrong reconfiguration action.

4.10.4 Modelling a Regional Tele-control System (*RTS*)

In Figure 50 is shown the SAN for the component *RTS* (see also Figure 35).

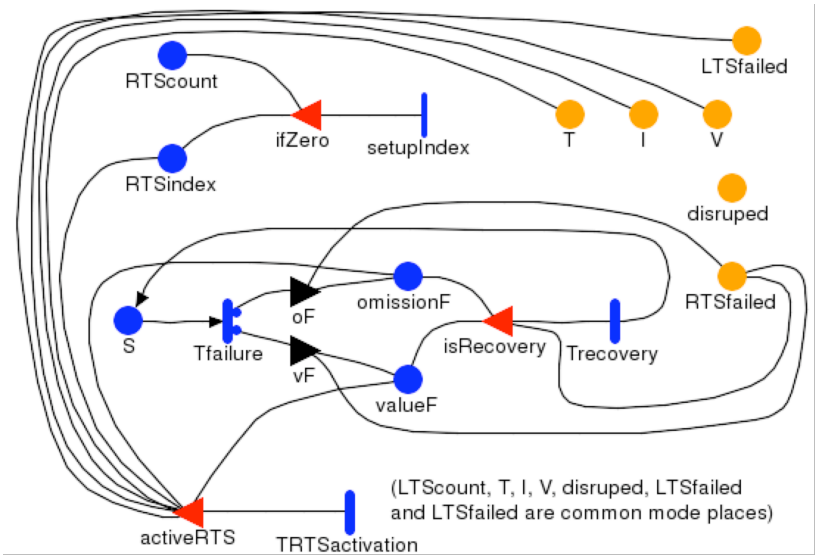


Figure 50: SAN for RTS

The extended place *LTSfailed* (common place)is used for the interactions between the replicas of *LTS* and *RTS*. The extended place *RTSfailed* (common place) is used for the interactions between the replicas of *RTS*. The activities *Tfailure* and *Trecovery* represent the time to failure and to recovery of the *RTS*. A failure of *RTS* may be an omission failure (case1) or a value failure (case2). When an omission failure occurs, the value of *RTSfailed–>Index(RTSindex–>Mark())–>value–>Mark()* is set to 1 by the gate *oF*. When a value failure occurs, the value of *RTSfailed–>Index(RTSindex–>Mark())–>omission–>Mark()* is set to 1 by the gate *vF*. When a repair occurs, these values are set to 0 by the gate *isRecovery*. The activity *TRTSactivation* represents the time that *RTS* spends to trigger the reconfiguration actions. This activity is enabled when there is not an omission failure and when the value of the function *RTSenable(LTSfailed, T, I, V, RTSindex->Mark())* is true. The result of the function *RTSenable()* depends on the values of the electrical parameters and on the topology of the controlled nodes and lines, as well as on the components *LTS* which are failed and on the type of their failures, i.e., on the marking of *LTSfailed*. When the activity completes, the following actions are performed by the gate *activeRTS* (CharA4 of Section 4.7):

RTSreconfigure(LTSfailed, T, RTSindex->Mark());
if (valueF->Mark()!=0) RTSchangeValue(T, RTSindex->Mark());
RLTSchangeValue(LTSfailed, T, RTSindex->Mark());
autoevolution(T, I, V);

The function *RTSreconfigure()* can change the part of the topology related to the region of the *RTS* with index *RTSindex->Mark()*. The functions *RTSchangeValue()* and *RLTSchangeValue()* perform the changes of the configuration due to the value failure of RTS and *LTS*, respectively. In both cases it's represented a potentially wrong reconfiguration action that, in the first case, may affect several controlled *LTS*.

4.10.5 Building an instance of the EPS

In Figure 51 it is shown how the atomic models for N_S, A_L, *LTS* and *RTS* can be replicated (using "Rep" operator [Sanders & Meyer 2001]) and composed (using "Join" operator [Sanders & Meyer 2001]) to obtain a part of the EPS model.

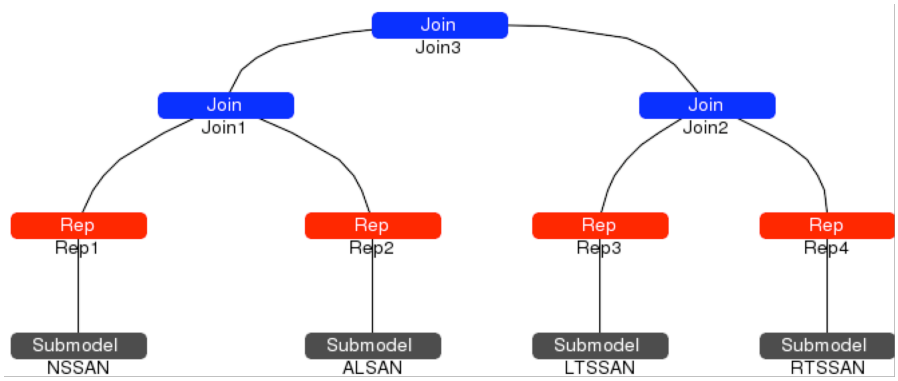


Figure 51: Part of the composed model for EPS

The number of replicas of the models of N_S, A_L, *LTS* and *RTS* are *n_S*, *n_A*, *n_LTS* and *n_RTS*, with *n_RTS* the number of regions (CharB1 of Section 4.7). The replicas interact through the places defined as common in each model. We remind that this anonymous replication becomes non anonymous thanks to the modelling mechanisms described in the first part of this section (CharB2 of Section 4.7).

4.11 Definition of the simulator EPSyS

As already discussed, we thought to assist the development of the generic modelling framework with a simulation approach to get insights on the involved phenomena, although under some restricted conditions. We first looked at tools already available, but found a lack of dedicated tools able to account for explicit infrastructures interactions in EPS analysis. While there are of course well-known and powerful EI simulators, and II simulators exist as well (with focus on some specific topics, most notably telecommunications), integrated simulators are still emerging. Tools addressing the interactions between different systems (infrastructures) are often specialized cases of tools addressing the problem to evaluate Complex Interconnected Systems. Those tools are designed to have enough modelling power to represent heterogeneous interconnected systems and to allow to reproduce the behaviour of specific subsystems. Their design aims to help to highlight the interdependencies and the interactions between modelled subsystems. The components of the modelled system are either represented in terms of their qualitative behaviour, or by attaching domain-specific tools, external models or simulators to increase the level of detail of the simulation. Unfortunately, most of those tools are still on-going research effort, or they reuse domain-specific simulator coupled with glue components, as done in EPOCHS [Hopkinson *et al.* 2003], which brings together the best pieces available, but also increases the overall simulator complexity both in terms of development and usage issue. Also the EU project IRRIIS (http://www.irriis.org/) is developing SimCIP (Simulation for Critical Infrastructure Protection), an agent-based simulation environment for controlled experimentation with a special focus on critical infrastructures interdependencies. Contacts between the CRUTIAL and IRRIIS projects are in place, to keep updated on the reciprocal research directions and developments in the (partially) similar application environments the two projects are addressing.

In addition to the difficulties in finding a mature, well assisted and easy to customize simulator for EPS systems, we decided to build our EPSyS simulator since our aim was

primarily to use it in conjunction with the modelling framework under investigation, for reciprocal benefits. Therefore, we needed to have the two evaluation methods sharing the same common knowledge and representation of the stochastic models capturing the behaviours and dynamics of the entities composing the EPS system. However, the development of EPSyS has been held highly modular so, if needed, it is possible to interface domain-specific simulators with some modules, in order to achieve a more realistic simulation. The detailed description of the EPSyS tool is presented in Deliverable D11. In the following we describe the stochastic models adopted in EPSyS.

4.11.1 EPSyS stochastic models

This section describes the stochastic models adopted in EPSyS. As already mentioned, they are basically the same as those adopted in the modelling framework and described in the previous Subsections. More precisely, the EI components and dynamics are represented in a more detailed way, while the II operations are simply captured through two functions which represent the overall reconfiguration they perform on the EI infrastructure.

4.11.1.1 Power flow model

The real input power at each node *i* is P_i, which is positive for the generators, negative for the loads and zero for the substations. The maximum power that a generator *i* can supply is $P_i^{\max}$ and the maximum power flow that a transmission line *l* can carry is $F_l^{\max}$. A line is overloaded if the power flow exceeds $F_l^{\max}$. The impedance of each line *l* is z_l.

The electric state of the transmission grid can be represented by the values of various electric parameters associated to each component of the grids: the voltage, the frequency, the angle, the active and reactive power flow. For obtaining information about the values of all these parameters when the state changes during the evolution of the power grid system, very difficult and extremely time-consuming or computationally intensive numerical problems should be solved.

is an extremely time-consuming or computationally intensive numerical problem. For example, full nonlinear equations and optimizations are needed even when only steady-state operative conditions are considered. Moreover, the random and cascading disruptions (or disturbances or contingencies) that may occur during the stochastic evolution of the system and the number of simulation batches needed to obtain statistically significant measures of interest require numerous solutions of such numerical problems. Solving repeatedly numerous time-consuming problems is computationally prohibitive for our purposes. For these reasons, EPSyS adopts some simplifying assumptions with the aim to study the power flow through the network. Following the same standard approach used in literature, the state and the evolution of the transmission grid are described by the distribution of the active power flows which are computed using a linear “DC” (direct current) load flow approximation of the AC system. This approximation is based on the following assumptions:

1. The electric transmission grid operates in steady-state conditions. Steady-state refers to power supply reaching a state wherein the voltage and the power flows are nearly constant along time (after at least 30 seconds of operation at a given load), i.e. have reached an equilibrium condition.
2. All voltage magnitudes are 1.0 per unit .
3. $\cos(\theta_i - \theta_j) \approx 1$ and $\sin(\theta_i - \theta_j) \approx \theta_i - \theta_j$, where θ_i is the voltage phases at node *h*.
4. Transmission line resistance is negligible.
5. One of the generators is defined as the *reference node* (slack node), which has voltage angle zero.

6. The total generated power is forced to exactly balance the total load demand:

$$\sum_{i \in G \cup L} P_i = 0 \qquad (1)$$

(power balance constraint), where G is the set of generators and L is the set of loads.

Under these assumptions, the equations for the "DC" power flow approximation can be derived from the standard AC circuit equations. They can be written as:

$$\begin{aligned} P &= B \bullet \Theta \\ F &= b \bullet A \bullet \Theta \end{aligned} \qquad (2)$$

where:

- $B = A^T \cdot b \cdot A$ is the *n×n susceptance matrix*, and A^T is the transpose of *A*,
- $b = diag(b_1, b_2, ..., b_m)$is the *m×m* diagonal matrix with each entry $b_l = 1/z_l$ being the susceptance of each transmission line *l*,
- $\Theta = (\theta_1, \theta_2, ..., \theta_n)^T$ is the *node voltage angle vector*,
- $P = (P_1, P_2, ..., P_n)^T$ is the *real power injection vector*,
- $F = (F_1, F_2, ..., F_m)^T$ is the *line power flow vector*.

Equations (2) give a linear relationship between the power flow F on the lines and the input power P at the nodes. The active power flow on the transmission lines are obtained in terms of the voltage phases from equation. The voltage phases are obtained from equation in terms of the input power at the nodes, using the zero angle of the reference node and allowing for the singularity of the matrix *B*, which has rank *n*-1 because of the constraint. The real power injected at the reference node is computed from equation (1).

4.11.1.2 Failure model of the transmission grid

A disruption is the unexpected failure or outage of a generator, power line or substation. Components affected by a disruption are out of service (disconnected from the grid). When a substation is out of service, all the lines connected to the substation are out of service as well. A *random disruption* may trigger *cascading disruptions* in the grid and *cascading failures* in the II. The propagation of a disruption can be stopped by the protections by isolating from the grid the component affected by the disruption. Causes of a cascading disruption can be events such as: excessive heating of a component due to overloads, failures in the II components (e.g., incorrect line trips due to hidden failures in the protections), etc. The components which are out of service can be put back in service only when the cause of the disruption is removed, for example, after the repair or replacement of the *damaged* component (restoration) or after the overload of the component is removed by II. Anyway, after a disruption of a component, a time (from a few seconds to hours) must pass before the component can be put back in service. The failure model is based on the following assumptions; several of them are also used in similar studies in the literature:

1. Disruptions for which the component can be put back in service in a few seconds are not considered.
2. When a random disruption occurs, the affected component is considered damaged. In this case, the repair is required to put back in service the component.

3. Random disruptions are statistically independent in space and time.
4. The time of occurrence of random disruptions of a generator, power line or substation (which are in service) can be deterministic or random with general distribution selected among the most common and realistic ones (such as exponential with rate λ_i).
5. The rate of occurrence λ_l of a random line disruption *l* is proportional to the length of the line.
6. The (random or cascading) disruption of a component propagates to a neighbour component *j* (causing a cascading disruption) with probability $p^{HF}(P_j)$ dependent on the power flow P_j through *j* and on the subsequent exposures to disruptions. The probability $p^{HF}(P_j)$ represents the failure of the protection. In this case the component *j* is considered damaged, i.e., repair is required to put back in service the component.
7. When the excessive heating of a line *l* due to overloads causes a (cascading) disruption of *l*, the line *l* is damaged (due to a failure of the protection) with probability $p^{HF}(F_l)$. Otherwise, the line is only disconnected by the protection.
8. For modelling the heating of a line, the spatial variation in the temperature along the line and the lose of heat of each element of the surface of the rod by radiation to the surrounding medium are not taken into account.
9. Transient instability, i.e., the disconnection of one or more generators because of loss of synchronism, has not been (explicitly) considered.
10. A big variation of power flow through a generator in a small interval of time (for example, due to a variation of the load) causes a (cascading) disruption. In this case, the generator *i* is damaged (due to a failure of the protection) with probability $P^{HF}(P_i)$, otherwise it is only disconnected by the protection.

From assumption 5, the parameters of the distributions of the time of occurrence of a random disruption of a line *l* are defined in such a way that $\lambda_l = \lambda L_l$ where λ is the constant failure rate for unit length and L_l is the length of the line. Overloads are caused by reactions to disruptions and reconfigurations triggered by II. For modelling the heating of a line *l* due to overloads, the same approach proposed in is considered, based on the assumption. The line temperature at time *t* is given by the simple approximated equation:

$$T_l(t) = e^{-\nu t}(T_l(0) - T_{el}(F_l)) + T_{el}(F_l) \qquad (3)$$

where *T*(0) is the initial temperature of the line and

$$T_{el}(F_l) = \frac{\alpha}{\nu}\frac{F_l^2}{V_l^2} + T_0 \qquad (4)$$

is the equilibrium temperature of *l* for $t \to \infty$, where T_0 is the temperature of the medium. The parameters α and ν are defined as α=0.239/($\rho c\omega^2\sigma$) and ν=*Hp*/($\rho c\omega$), supposing that the transmission line has constant area of cross-section ω, perimeter *p*, electrical conductivity σ, density ρ, specific heat *c* and surface conductance *H*, with $H = 8\times10^{-5}(u/d)^{1/2}$, for a turbolent flow of air with velocity *u* perpendicular to a circular cylinder of diameter *d*. When $F_l = F_l^{\max}$ the temperature $T_l(t)$ converges to the critical temperature T_{dl}, for which the line *l* sags and trips. A (cascading) disruption of *l* due to the power flow F_l, with $F_l \geq F_l^{\max}$, occurs when the temperature reaches T_{dl} at some time t_{dl} (measured from the moment when the power flow through *l* has changed). The time to disruption t_{dl} of line *l* is given by:

$$t_{dl}(F_l) = -\frac{1}{v}\ln\frac{T_{dl} - T_{el}(F_l)}{T_0 - T_{el}(F_l)}$$

The generators cannot fail due to an overload, because produced power cannot exceed a given threshold. For the assumption 10, a generator can fail due to a power flow variation greater than a given threshold $\Delta P_i^{\max}$ in a given $\Delta t_i^{\max}$ time interval. Thus, a (cascading) disruption of a generator *i* occurs when:

$$\left|\frac{P_i(t_2) - P_i(t_1)}{t_2 - t_1}\right| > \left|\frac{\Delta P_i^{\max}}{\Delta t_i^{\max}}\right|$$

where P_i^{t2} is the new power flow injected on *i* at time t_2 and P_i^{t1} is the previous one injected on *i* at time t_1, with $t_1 \leq t_2$.

Repair of a damaged component is considered in the system model, but it is not yet implemented in the simulator. Considering the transient instability requires a significantly more detailed model. For simplifying the model, in this version of the simulator, the probability of loss of synchronism has been considered equal to zero.

4.11.1.3 II model

For simplifying the II model, the effect on the transmission grid of generation redispatch, load shedding or grid reconfiguration is only considered, and the details of the different components of II are not taken into account. The behaviour of II is structured in two levels corresponding to LCS and RTS/NTS. Each level is characterized by an activation condition, a reaction delay and a (dispatch, shedding and) reconfiguration strategy ($\mathcal{RS}$). The activation condition (defined as a simple predicate) specifies the grid events that enable the reaction of a specific level of II. Different events or sequences of events can enable different reaction levels. The reaction delay models the overall computation and application time needed by II to apply a reconfiguration, which can be considered immediate for local decisions. The reconfiguration strategy $\mathcal{RS}$ defines how the configuration of EI changes when II reacts to a disruption. For each level, a different reconfiguration function is modelled:

- $RS_1()$, to represent the effect on the complete transmission grid of the local and fast reactions to a disruption (LCS).
- $RS_2()$, to represent the effect on the complete transmission grid of the global and slower reaction to a disruption (RTS/NTS).

Both these functions receive in input the state of the EI at the time immediately before the occurrence of the disruption and output the new values for P and F, which are the result of the reaction of II to the disruption. The following simplifying assumptions are considered:

1. The output values of $RS_1()$ and $RS_2()$ for P and F satisfy the power flow equations.
2. The reaction to a disruption represented by $RS_1()$ is "worse" (from the point of view of the tradeoff between voltage quality and costs) than the reaction represented by $RS_2()$, being based on a local view of the state of the system and requiring only a few seconds to react to a disruption.
3. The times to trigger $RS_1()$ and $RS_2()$ are $T_1 = 0$ and $T_2 > 0$, respectively.

The definition of the functions $RS_1()$ and $RS_2()$ depends on the policies and algorithms adopted by II. For a given load power demand, the power flow equations do not have a unique solution. There are many combinations of generator powers to satisfy a given load demand. In the standard approaches adopted in literature the solution to this equations system is formulated as an optimization to minimize the change in generation or load shedding subject to the system constraints. Therefore, a possible definition of the function $RS_2()$ is given by the solution (values for P and F) of power flow equations while minimizing the simple cost function:

$$C_2 = \sum_{i \in G} \left| P_i - P_i^0 \right| + W_L \sum_{i \in L} \left| P_i - P_i^0 \right|$$

with the following constraints:

$$\begin{aligned} & 0 \le P_i \le P_i^{\max}, i \in G, \\ & P_i^0 \le P_i \le 0, i \in L, \\ & -kF_l^{\max} \le F_l \le kF_l^{\max}, l \in F \end{aligned}$$

where P_i^0 is the injected power immediately before the occurrence of the disruption that triggers $RS_2()$. The parameter W_L is the cost for load shedding, which is set to an high value in order to force the generation dispatch first. The line overload parameter *k* represents either the risk of adverse reaction of II (when *k*<1), or a risk of taking a reaction of II (when *k*>1). In C_2, the cost to adjust the generators is the same and the loads have the same priority to be served. The function C_2 aims at performing the least possible modifications of the electric state of the grid, with respect to the electric state immediately before the occurrence of the disruption that triggered $RS_2()$. Different cost functions can be considered for $RS_2()$, such as:

$$C_2' = \sum_{i \in G} \left| P_i - P_i^0 \right| + W_L \sum \left| P_i - P_i^0 \right| + W_z z$$

with the following constraint added to previous ones:

$$-z \le F_l \le z, l \in F$$

This function minimizes a tradeoff between the change in generation, the load shedding and the maximum load through a line. The parameter W_z is the price for maximum load through a line. It can be set to a value such that $W_L >> W_z >> 0$ in order to force the reduction of the maximum load through a line before the generation dispatch, or to a value such that $W_z < 1$ (and $W_L >> 0$) in order to force the generation dispatch first. It is important to note that the definitions considered for $RS_2()$ are based on the same optimization problems that can be solved (on-line or off-line, if possible) by II to react to a disruption. On the contrary, for defining

$RS_1()$ the algorithms adopted by II are not taken into account explicitly, but only their possible and supposed effects on the overall system are considered in terms of the solutions to the power flow equations. A possible definition of the function $RS_2()$ is given by the following steps:

1. For a given load power demand and distribution of generation P, the solution F of

power flow equations is obtained, if it exists. In this case redispatch or load shedding are not needed, but the power flow through the lines increases and it can produce overload, especially if constraint 12 is not considered or if *K*>1.

2. Otherwise (if redispatch or load shedding are required) the values for P and F are obtained by solving the optimization problem using the $C_2^{'}$ cost function, by removing the line overload and shed prevention constraints and, in general, using a "worse" choice of weights. A worse choice, is, for example, $W_P \approx W_L, W_P - W_z$ which do not prevent line overloads and not avoid any shedding; as another example $W_P - W_z, W_L$ causes load shedding in response to increased power request.

4.11.1.4 Failure model of II

Following the approach proposed for the hidden failure model of the protections, the probability $p^{HF}(P)$ that a protection fails is low when the power flow is below the component limit P^{CL}, and increases linearly to 1 when the component flow is 1.4 times the P^{CL}:

$$p^{HF}(P) = \begin{cases} p^{HF}, if \cdots P < P^{CL} \\ \dfrac{1 - P^{HF}}{0.4 P^{CL}} P + \dfrac{1.4 P^{HF} - 1}{0.4}, if \cdots P^{CL} < P \\ 1 \cdots otherwise \end{cases}$$

Moreover, the hidden failure probability P^{HF} reduces to zero after the first exposure to disruption. Failures of the II components are considered in the model, including transient and permanent omission failures, time failures, value failures and byzantine failures. Here the focus is on the failures and not on their causes (internal HW/SW faults, malicious attacks, etc.). The dependencies from EI to II (cascading failures) could be also considered, for example handling the case that a blackout shrinks the performance of II. In the current implementation only omission and timing failures of II are considered.

4.12 Discussion

We have discussed the preliminary approaches to the analysis and quantitative evaluation of interdependencies between the electric and information infrastructures. The two directions under investigations have been discussed and motivated. On one side, we are working on the definition of template models capturing the structure, dynamics and failure propagation phenomena of the EPS entities, to be used to build complete models in generic control system scenarios. The prominent aspects of the EPS modelling framework have been identified, and initial development of models for the major EPS entities have been performed using the tool Möbius and Stochastic Activity Networks (SAN). On the other side, we are developing a simulator to assist the definition of the modelling framework. A first version of the simulator is available and used in a few artificial scenarios to acquire a better knowledge of the impact of interdependencies through properly identified metrics. The results obtained so far are in line with the expectations and so fruitfully exploited in the modelling framework. In addition, the developments of the two methods to assess the consequences of failures in power systems are also very useful to be exploited for cross validation among the methods themselves.

5 CONCLUSIONS

In this deliverable, we presented the progress achieved so far by CRUTIAL toward the development of a modelling framework aimed at the description of interdependencies-related failures and the assessment of their impact on the dependability and security of the services delivered by the electricity power systems. The proposed framework includes 1) qualitative models characterizing the occurrence of cascading, escalating and common cause failures, and 2) quantitative assessment models taking into account the internal structure of the electrical infrastructure and the associated information infrastructures supporting monitoring, communication and control functionalities.

Significant contributions and extensions have been obtained in the course of the second year of the project considering both types of models.

As regards the qualitative models, the main contributions include: 1) the development of unified models considering accidental and malicious threats in a integrated way, 2) the proposal of a formal setting for describing interdependencies-related failures and 3) the elaboration of compositional modelling framework based on GSPNs that is well suited to facilitate the generation of the qualitative models and to favour their reusability in other contexts. At the current stage, the formalisation and the compositional modelling framework do not distinguish between the notions of real and apparent states that appear in the unified model. Future extensions will consist in integrating such a distinction and using concrete examples for illustration based on selected scenarios from those that have defined in WP1 and rediscussed in this deliverable.

Concerning the quantitative assessment models, we have continued the development of template models capturing the structure, dynamics and failure propagation phenomena of the electrical power system entities, to be used to build complete models in generic control system scenarios. The formal definition of these models is based on the stochastic activity networks (SAN) formalism. We also concentrated on the development of an ad-hoc simulator for EPS systems to support the definition and refinement of these template models by providing quick feedbacks on particular interdependencies related behaviour, although under some restricted system conditions. In addition, the possibility of cross-validation between the two methods has been among the motivations for pursuing the two research directions for quantitative assessment. A quite rich research agenda is currently planned for the last year of the project, on both the approaches. On the modelling side, refinements and completion of the template models are the next steps. Affording complete case studies, both artificial and taken from the CRUTIAL control system scenarios derived in WP1, by composition of these template models in accordance with specific metrics of interest, is then the natural follow up. On the other side, the simulator will be advanced in terms of: i) refinement/additions of the functionalities of the information infrastructures, ii) implementation of the repair process which brings back in operation permanently faulty components; iii) inclusion of more sophisticated failure patterns; iv) usage in more complex and realistic case studies.

REFERENCES

[Ajmone Marsan *et al.* 1995] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli and F. G., *Modelling with Generalized Stochastic Petri Nets,* John Wiley, 1995.

[CESI RICERCA 2006a] CESI RICERCA, *Control system scenarios with interdependencies. INPUT to CRUTIAL WP1 rev 0 - Draft*, CESI RICERCA S.p.A., Grid and Infrastructures Department, August 2006a.

[CESI RICERCA 2006b] CESI RICERCA, *Description of Hierarchical Control Schema for the Power Systems and Critical Aspects of their Evaluation. Input to CRUTIAL WP1*, CESI RICERCA S.p.A., Network and Infrastructures Department, May 2006b.

[Chiola *et al.* 1993] G. Chiola, C. Dutheillet, G. Franceschinis and S. Haddad, "Well-formed Couloured nets for symmetric modelling applications", *IEEE Transactions on Computers*, 42 (11), pp.1343-1360, November 1993.

[Clark *et al.* 2001] G. Clark, T. Courtney, D. Daly, D. D. Deavours, S. Derisavi, J. M. Doyle, W. H. Sanders and P. Webster, "The Möbius Modeling Tool. In pnpm, p. 0241, " in *9th international Workshop on Petri Nets and Performance Models (PNPM'01),* p.2041, 2001.

[Garrone *et al.* 2007] F. Garrone, C. Brasca, D. Cerotti, D. Codetta Raiteri, A. Daidone, G. Deconinck, S. Donatelli, G. Dondossola, F. Grandoni, M. Kaâniche and T. Rigole, *Analysis of New Control Applications. Critical Utility InfrastructurAL Resilience.* Project co-funded by the European Commission within the Sixth Framework Programme. Deliverable D2, 2007.

[Hopkinson *et al.* 2003] M. K. Hopkinson, R. Giovanini, X. Wang, K. P. Birman, D. V. Coury and J. S. Thorp, *"EPOCHS: Integrated Cots Software for Agent-Based Electric Power and Communication Simulation, "* Winter Simulation Conference,, New Orleans, USA, 2003.

[Kaâniche *et al.* 2007] M. Kaâniche, S. Bernardi, A. Bobbio, C. Brasca, S. Chiaradonna, D. Codetta Raiteri, F. Di Giandomenico, G. Dondossola, G. Franceschini, F. Garrone, A. Horváth, K. Kanoun, J.-C. Laprie, P. Lollini and J. Sproston, *Methodologies Synthesis.* Critical Utility InfrastructurAL Resilience. Project co-funded by the European Commission within the Sixth Framework Programme. Deliverable D3, January 2007.

[Laprie *et al.* 2007] J.-C. Laprie, K. Kanoun and M. Kaâniche, "Modeling Interdependencies between the electricity and Information Infrastructures", in *26th International Conference on Computer Safety, Reliability and Security (SAFECOMP'2007),* (Nuremberg, Germany), pp.54-67, Springer, 2007.

[Pourbeik *et al.* 2006] P. Pourbeik, P. S. Kundur and C. W. Taylor, "The Anatomy of a Power Grid Blackout", *IEEE Power & Energy Magazine* (September/october), pp.22-29, 2006.

[Rinaldi *et al.* 2001] S. M. Rinaldi, J. P. Peerenboom and T. K. Kelly, "Identifying, Understanding, and Analyzing Critical Infrastructure Interdependencies", *IEEE Control Systems Magazine* (December), pp.11-25, 2001.

[Sanders & Meyer 2001] W. H. Sanders and J. F. Meyer, "Stochastic activity networks: Formal definitions and concepts", in *Lectures on Formal Methods and Performance Analysis. Lecture Notes in Computer Science* 2090, pp.315-343, Springer-Verlag, 2001.

[US-Canada 2004] US-Canada, *Power System Outage Task Force — Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations*, April 2004.