

STRUMENTI DI ANALISI DEL GUASTO NEI SISTEMI DI REMOTE-BANKING

Crescenzo Gallo⁽¹⁾

*Università di Foggia (Italy)
Dipartimento di Scienze Economiche, Matematiche e
Statistiche
Via IV Novembre, 1 - 71100 Foggia*

Abstract

I canali distribuiti innovativi hanno un potenziale impatto su tutte le fasi del rapporto tra cliente ed intermediario finanziario. La banca virtuale modifica i processi e le modalità di produzione mediante la tecnologia, ricercando efficienza ed efficacia con le forme di remote/home banking. Lo scopo di questo lavoro è quello di studiare attraverso l'uso di opportune tecniche (FTA, Decision Table) i modi di guasto e i loro effetti sul sistema di remote Banking per consentire la modellazione del sistema informativo secondo i possibili scenari creati dalle varie esigenze del mercato e dalla clientela.

Keywords: banca virtuale, remote-banking, fault-tree analysis, decision-table.

¹ email: c.gallo@unifg.it

1. Introduzione

La quantificazione dei costi di realizzazione di un sistema di remote banking dipende dalle scelte architettoniche (hardware, reti, applicazioni), dalle competenze applicative disponibili, dalle funzionalità che si intendono realizzare, e dal livello di integrazione con il sistema tradizionale. Come conseguenza immediata dell'aumento della qualità e della complessità dei servizi offerti, crescono i costi della tecnologia a supporto dell'implementazione del canale di distribuzione virtuale [15]. Il 70% dei costi di investimento dei servizi finanziari erogati via Internet e più del 40% dei costi di gestione hanno origine tecnologica [3]. La valutazione è complessa, perché sono in gioco numerosi fattori quali la tecnologia, l'organizzazione, i servizi finanziari, il cliente, il mercato, la concorrenza, l'informatica. Un modello di riferimento è rappresentato da una banca che utilizza Internet come un canale aggiuntivo rispetto alla filiale. Il sistema di Internet banking viene integrato con il sistema tradizionale con i vincoli che ne derivano, come la disponibilità del servizio che dipende direttamente dall'operatività del sistema centrale [14].

2. I modelli organizzativi e tecnologici

Il termine *banca virtuale* indica un'entità finanziaria che vende servizi finanziari alla clientela tramite le tecnologie dell'informazione e della comunicazione, senza ricorrere al personale di filiale o riducendolo in maniera elevata rispetto ad un sistema tradizionale [10]. Dal punto di vista della distribuzione, si è in presenza quindi di una dematerializzazione dei canali tradizionalmente usati dalle banche, vale a dire sportelli e agenzie [8]. Quando si parla di banca virtuale, si evoca una condizione nell'ambito della quale, attraverso l'uso di nuove tecnologie informatiche, un cliente riceve gli effetti di un'attività di tipo bancario in assenza degli elementi tipicamente costitutivi della banca stessa; siamo quindi di fronte ad una destrutturazione della banca tradizionale con il prevalere delle relazioni a distanza (remote banking)

con la clientela [1,6,7]. Tre sono gli aspetti rilevanti per le analisi costi-benefici: il ruolo fondamentale delle tecnologie informatiche, il contesto in cui avvengono le transazioni e le infrastrutture utilizzate, l'importanza dei sistemi distributivi remoti [4].

3. Gli strumenti di analisi

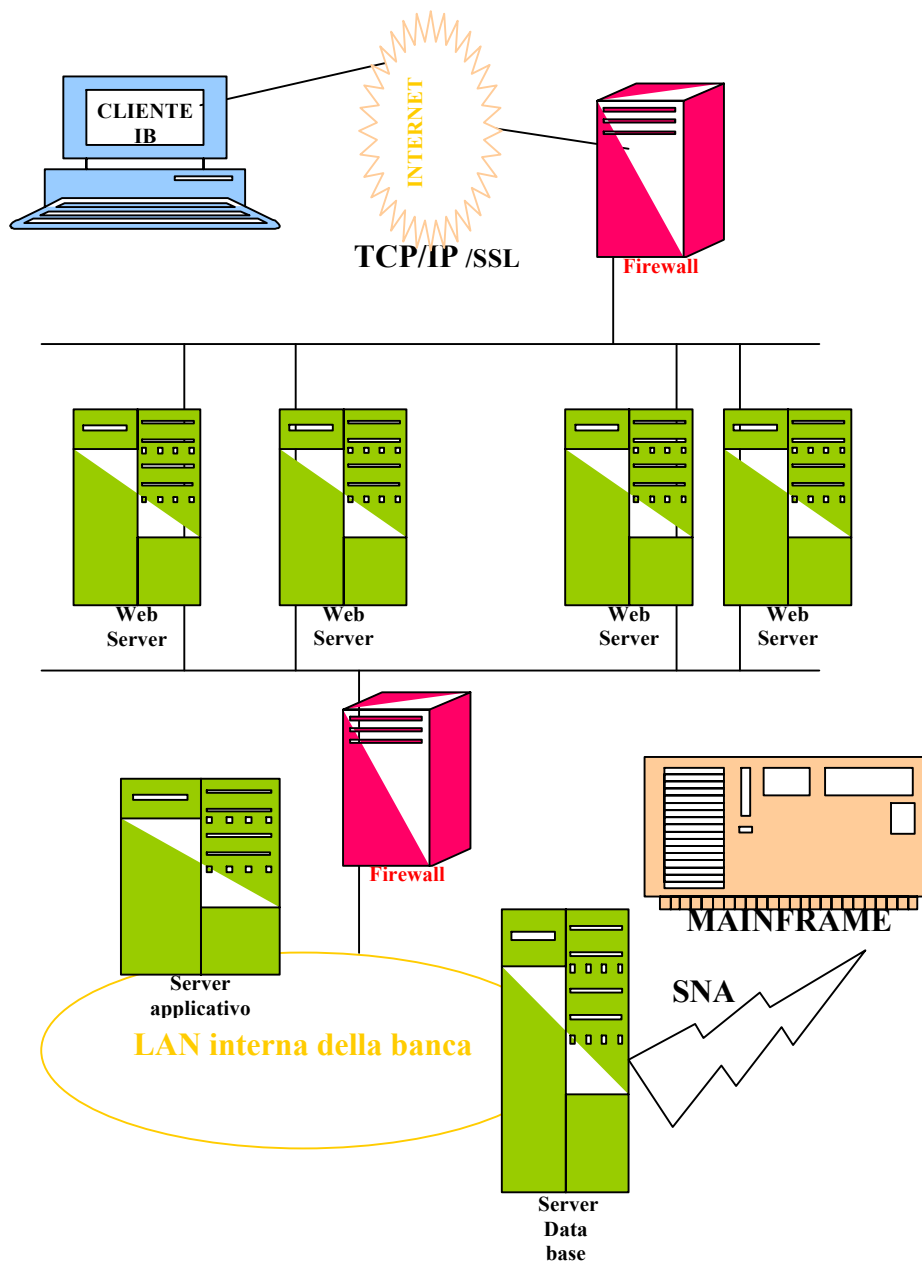
La formalizzazione di un sistema di remote-banking (sia dal punto di vista prettamente tecnologico che dall'aspetto di interrelazione con l'utente finale ed il back-end informatico centralizzato) richiede l'individuazione dei componenti fondamentali che lo costituiscono, quali:

- sistemi di elaborazione e conservazione centralizzata delle informazioni (mainframe) [13];
- sistemi di deployment dei dati centralizzati verso la periferia ed implementazione delle "business rules"(data server);
- sistemi di interfacciamento verso l'utente finale e di presentazione delle informazioni (web server) nel classico schema a tre livelli ("three-tier") universalmente adottato negli ambienti ad alta complessità elaborativa [2], insieme ai sistemi intermedi di controllo degli accessi e di gestione della sicurezza (firewall) indispensabili in un ambiente critico come quello bancario (vedi figura 1).

L'analisi di sistemi complessi come quelli di remote-banking richiede quindi l'uso di strumenti in grado di formalizzare da un lato i componenti principali indicati e dall'altro le relazioni tra gli stessi che aiutino a studiare ed evidenziare le possibilità di guasto dei singoli elementi e quindi del sistema nel suo complesso [5].

Lo strumento di analisi individuato che si presta in maniera ottimale agli obiettivi del nostro lavoro è quello degli "alberi di guasto" o "fault tree", con la relativa tecnica di "fault-tree analysis".

Fig. 1 - Internet banking: architettura hardware di riferimento.



L'analisi degli alberi di guasto o FTA (*Fault Tree Analysis*) è stata inizialmente introdotta per lo studio della sicurezza di sistemi di controllo missilistici; successivamente, questo metodo è stato utilizzato sempre più diffusamente nell'industria aerospaziale [9] e nucleare e in generale per lo studio di sistemi complessi di grosse dimensioni. Essa è particolarmente adatta per l'analisi di sistemi altamente ridondanti, mentre per sistemi particolarmente vulnerabili a singoli guasti che possono provocare un incidente è preferibile utilizzare tecniche di tipo diverso come FMEA (*Failure Modes and Effects Analysis*) o HAZOP (*Hazard and Operability Analysis*).

L'uso della FTA richiede una conoscenza dettagliata del funzionamento del sistema, dei modi di guasto dei suoi componenti e dei loro effetti. Il tempo ed il costo di una FTA dipendono dalla complessità del sistema in esame e dal livello di risoluzione dell'analisi: un sistema relativo ad un processo semplice può essere risolto da un'analista in un giorno, mentre un sistema complesso che richiede lo sviluppo di più alberi di guasto può impegnare un gruppo di esperti anche per mesi.

Questo metodo può essere applicato sia durante la fase di progettazione di un nuovo sistema sia in fase di verifica di uno già esistente, allo scopo di migliorarne la sicurezza poiché permette di individuare importanti caratteristiche come i punti deboli del sistema, false ridondanze o gli effetti di un dato componente sull'affidabilità complessiva.

La tecnica degli alberi di guasto richiede la decomposizione del sistema in un diagramma logico, detto *albero di guasto*, in cui certi eventi primari conducono ad uno specifico evento che rappresenta l'avaria totale del sistema, detto *Top Event* poiché si trova in cima all'albero di guasto. Iniziando dal *Top Event*, l'albero di guasto è costruito ramificandosi verso livelli più bassi costituiti da eventi intermedi che potrebbero determinare il *Top Event*; usando i comuni operatori logici AND, OR, NOT si ricostruisce la sequenza degli eventi fino ad arrivare agli eventi di base di cui sono note le probabilità di accadimento. Una volta che l'albero di guasto è stato costruito, la probabilità del *Top Event* può essere determinata calcolando la

probabilità degli eventi intermedi partendo dal basso dell'albero e procedendo verso l'alto.

Per evento di base, allora, si intende un qualunque guasto, o un qualunque errore umano, che porti alla rottura o al malfunzionamento di un componente del sistema complesso in esame.

Un albero di guasto è un albero a struttura rovesciata, nel quale, muovendoci dall'alto verso il basso, troviamo come primo componente il *Top Event* e, scendendo di livello in livello, arriviamo alle foglie dell'albero che sono costituite dagli eventi di base.

Per rappresentare la sequenza di eventi che devono verificarsi per arrivare all'accadimento del *Top Event*, sono presenti nell'albero, a vari livelli, dei nodi costituiti da *gates* che rappresentano graficamente le relazioni che intercorrono tra i rami che convergono in ciascun nodo. I *gates* maggiormente usati sono essenzialmente **OR**, **AND**, **NOT**, **NAND**, **NOR**, **EOR** (o **XOR**).

Il metodo sicuramente migliore per la risoluzione di un albero di guasto è quello che utilizza la sua *espressione booleana* per definire la *funzione di guasto* caratteristica dell'albero, il cui valore rappresenterà lo stato del sistema in esame.

La possibilità di esprimere la funzione di guasto mediante un'espressione booleana è dovuta al fatto che si prendono in considerazione unicamente sistemi binari, sistemi cioè in cui è possibile rappresentare lo stato di ogni componente, o di ogni evento di base, mediante delle variabili logiche binarie, variabili dalle quali poi dipenderà la funzione di guasto stessa. L'espressione booleana dell'albero di guasto si presta particolarmente bene alla determinazione dei *Cut-Set* termine con il quale si indicano tutte quelle particolari concatenazioni di eventi che portano al verificarsi del *Top Event*. All'interno dell'insieme dei *Cut-Sets* è possibile definire un particolare sottoinsieme che è quello dei *Minimal Cut-Sets* (più brevemente *Mincut*). I *Minimal Cut-Set* sono di particolare importanza nell'analisi con gli alberi di guasto, e per questo risulta fondamentale la loro determinazione, poiché la semplice variazione dello stato di un evento appartenente ad un *Minimal Cut-Set* comporta infatti, che il *Minimal Cut-Set* in questione non possa più essere considerato un *Cut-Set* dell'albero di guasto che stiamo analizzando. Intervenendo quindi su un qualsiasi componente facente capo ad un *Minimal Cut-Set* possiamo di fatto migliorare l'affidabilità del sistema in esame.

La funzione di guasto, viene espressa mediante una *somma di prodotti* (*sum of products* o, più semplicemente, *s-o-p*). Espressa in questa particolare forma, qualora anche uno solo degli addendi di tale somma assuma valore 1, ovvero qualora tutti i fattori di un prodotto assumano valore 1 contemporaneamente, comporterà il verificarsi del *Top Event*. Basandoci su questa considerazione, e ricordando la definizione di *Cut-Set* si afferma quindi che ogni addendo della *somma di prodotti* rappresenta un *Cut-Set* dell'albero di guasto; se inoltre gli addendi sono tutti indipendenti tra loro, ovvero se nessuno di essi ne implica un altro, potremo affermare che tali addendi rappresentano i *Minimal Cut-Set* dell'albero di guasto

Una volta ricavata l'espressione booleana dell'albero nella forma di *somma di prodotti* è possibile valutare l'affidabilità R del sistema rappresentato dall'albero di guasto [12]. Prendendo in considerazione le possibilità di accadimento degli eventi di base e del *Top Event*, si può scrivere la seguente relazione:

$$P(X_s = 1) = 1 - R = \text{funzione di guasto}$$

$X_s = (\text{stato del sistema})$

$\Phi = (\text{funzione di guasto})$

$x_i = (\text{stato del componente } i - \text{esimo})$

La possibilità di guasto del sistema, ovvero la possibilità che la funzione di guasto assuma valore 1, può essere ricavata direttamente dalla *s-o-p* utilizzando le seguenti relazioni:

- Nel caso di eventi compatibili:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

- Nel caso di eventi incompatibili:

$$P(A \cup B) = P(A) + P(B)$$

- Nel caso di eventi stocasticamente indipendenti:

$$P(A) = P(A | B)$$

$$P(A \cap B) = P(A) \cdot P(B)$$

- Nel caso di eventi stocasticamente dipendenti:

$$P(A \cap B) = P(A | B) \cdot P(B) = P(B | A) \cdot P(A)$$

$$P(A) = P(A | B) \cdot P(B) / P(B | A)$$

Esiste anche un altro metodo per la determinazione della funzione R : si tratta di rielaborare la $s-o-p$ al fine di ottenere una **somma disgiunta di prodotti** ($d-s-o-p$). In generale due addendi della somma di prodotti si dicono disgiunti se si escludono mutuamente, ovvero se esiste almeno una variabile x_i che è presente non negata in un addendo e negata nell'altro. Una volta ricavata questa formulazione della funzione di guasto è ancora più semplice ottenere l'espressione dell'affidabilità del sistema: infatti si può notare che se S_i ed S_j sono due addendi della $d-s-o-p$ avremo $P\{S_i \cap S_j\} = 0$ e quindi l'espressione della probabilità della funzione di guasto si semplificherà assumendo la seguente forma:

$$P(X_s = 1) = P\left(\bigcup_{i=1}^n S_i\right) = \sum_{i=1}^n P(S_i)$$

dove con n indichiamo il numero degli eventi di base.

L'espressione della $d-s-o-p$ è molto semplice e porta quindi a preferire il metodo di calcolo dell'affidabilità basato sulla $d-s-o-p$. Allo scopo di ricavare la $d-s-o-p$ dell'albero di guasto in esame sono stati proposti molti algoritmi risolutivi fra i quali uno dei più efficaci è costituito dalla *Decomposizione di Shannon*. La possibilità di guasto può essere espressa nella seguente forma:

$$P(X_s = 1) = \sum_{i=1}^n P(S_i = 1) - \sum_{i=2}^n \sum_{j=1}^{i-1} P(S_i \cdot S_j = 1) + \dots (-1)^{n-1} P\left(\prod_{i=1}^n S_i = 1\right)$$

Nel caso in cui si abbia a che fare con sistemi di questo tipo, si può definire un intervallo al quale apparterrà il valore esatto della possibilità di guasto, ovvero stabilire un limite superiore ed un limite inferiore per la possibilità di guasto stessa. Tale intervallo si stabilisce considerando che un numero dispari di termini rappresenta un limite superiore per la possibilità di guasto, mentre un numero pari rappresenta un limite inferiore; considerando unicamente i primi due termini dell'espressione precedente si ottiene:

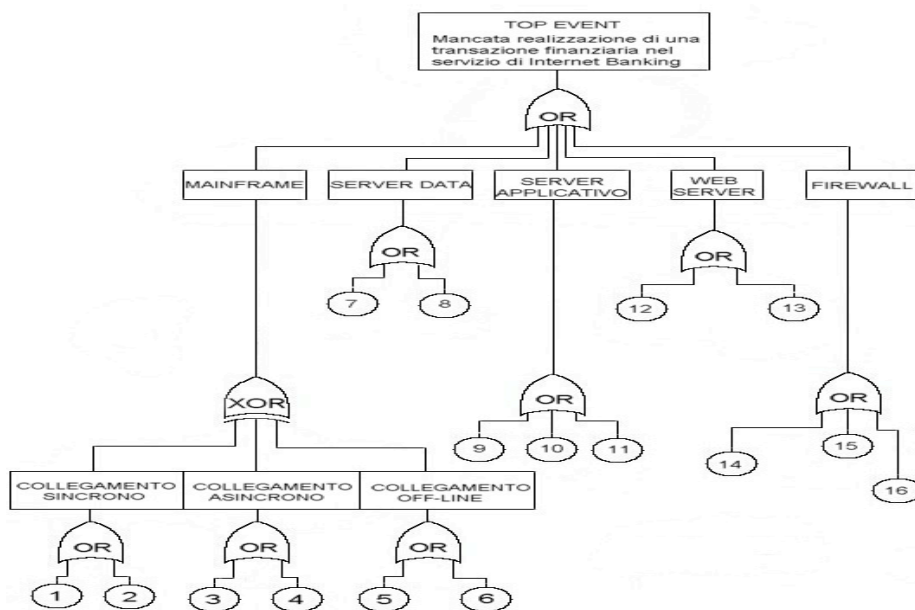
$$P(X_s = 1) \leq \sum_{i=1}^n P(S_i = 1)$$

$$P(X_s = 1) \geq \sum_{i=1}^n P(S_i = 1) - \sum_{i=2}^n \sum_{j=1}^{i-1} P(S_i \cdot S_j = 1)$$

Secondo le disuguaglianze di Bonferroni, essi rappresentano, almeno per alberi *coerenti* (alberi cioè in cui siano presenti unicamente gates di tipo *AND* o di tipo *OR*) una buona approssimazione del valore effettivo di *R*; per alberi *non-coerenti* si deve considerare un numero di termini maggiore.

La nostra indagine si è rivolta in particolare ad un sistema di remote-banking via Internet che abbiamo analizzato e formalizzato secondo la FTA come evidenziato nell'albero di guasto di figura 2.

Fig. 2 - Albero di guasto di un sistema di Internet banking.



A tale rappresentazione si perviene mediante un'analisi preliminare delle combinazioni di eventi di guasto tramite la tavola decisionale [11] di figura 3.

Fig. 3 - Tavola decisionale per l'analisi di un sistema di Internet banking.

DECISION TABLE																																	
MALFUNZIONAMENTO DEL:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	
MAINFRAME	F	T	T	T	F	F	F	F	T	T	T	T	T	T	F	F	F	F	T	T	T	T	F	F	F	F	F	F	F	F	T	F	
SERVER DATA	T	F	T	T	F	T	T	T	F	F	T	T	T	F	T	F	F	F	T	F	F	F	T	F	F	T	T	T	F	F	T	F	
SERVER APPLICATIVO	T	T	F	T	T	F	T	T	F	T	T	F	F	T	F	F	T	F	F	F	T	F	F	T	F	F	T	F	T	F	T	F	
WEB SERVER	T	T	T	F	T	T	F	T	T	F	T	F	T	F	F	F	F	T	F	F	F	T	F	F	T	F	F	T	F	T	F	T	F
FIREWALL	T	T	T	F	T	T	F	T	T	F	T	F	T	F	F	F	F	F	T	F	F	F	T	F	F	T	F	T	F	T	T	F	
5010-SE COLLEGAMENTO CLIENT	1- BLOCCO SISTEMA CLIENT	*	*	*	*					*	*	*	*	*	*	*							*	*	*	*						*	
	2- BLOCCO SISTEMA HOST	*	*	*	*					*	*	*	*	*	*	*							*	*	*	*						*	
5010-SE COLLEGAMENTO DELL'HOST	3- INDISPONIBILITA' DELL'HOST	*	*	*	*					*	*	*	*	*	*							*	*	*	*						*		
	4- OTTENIMENTO DI DATI NON AGGIORNATI	*	*	*	*					*	*	*	*	*	*							*	*	*	*						*		
5010-SE COLLEGAMENTO DATI NON AGGIORNATI	5- PRELEVAMENTO DATI NON AGGIORNATI	*	*	*	*					*	*	*	*	*	*							*	*	*	*						*		
	6- MANCANZA DI UN MECCANISMO DI SINCRONIZZAZIONE	*	*	*	*					*	*	*	*	*	*							*	*	*	*						*		
5010-SE COLLEGAMENTO MAINFRAME	7- INDISPONIBILITA' MAINFRAME	*	*	*	*					*	*	*	*	*	*				*	*	*	*								*			
	8- INDISPONIBILITA' COLLEGAMENTO VIA SNA	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
5010-SE COLLEGAMENTO SERVER DATA	9- INDISPONIBILITA' SERVER DATA	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
	10- MANCATA INTERCONNESSIONE CON IL SERVER	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
5010-SE COLLEGAMENTO ACCESSI	11- ELEVATO NUMERO DI ACCESSI	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
	12- ELEVATO NUMERO DI ACCESSI	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
5010-SE COLLEGAMENTO INTERCONNESSIONE	13- MANCATA INTERCONNESSIONE	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
	14- INSERIMENTO ERRATO NOME UTENTE	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
5010-SE COLLEGAMENTO PASSWORD	15- INSERIMENTO ERRATO PASSWORD	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		
	16- INSERIMENTO ERRATO P.I.N.	*	*	*	*					*	*	*	*	*	*			*	*	*	*		*	*	*	*	*	*	*	*	*		

Per applicare in maniera efficiente la tecnica della FTA al nostro albero di guasto si è reso necessario operare una opportuna sintesi dell'albero in questione come riportato in figura 4

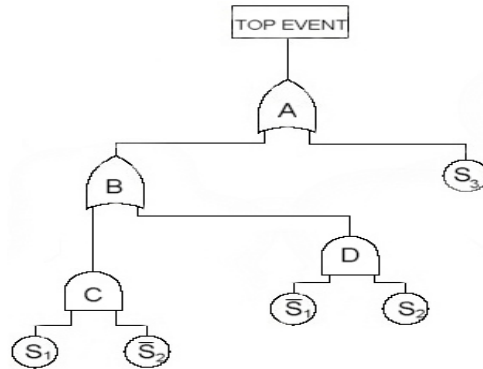
Fig. 4 - Albero sintetizzato.



Dove la sua espressione algebrica viene generata dalla seguente espressione riportata in figura 5:

Fig. 5 - Espressione algebrica dell'albero di guasto.

$$\begin{aligned} X_s &= S_1 \cap \bar{S}_2 \cup \bar{S}_1 \cap S_2 \cup S_3 \\ &= S_1 \otimes \bar{S}_2 \oplus \bar{S}_1 \otimes S_2 \oplus S_3 \\ &= S_1 \cdot \bar{S}_2 + \bar{S}_1 \cdot S_2 + S_3 \end{aligned}$$



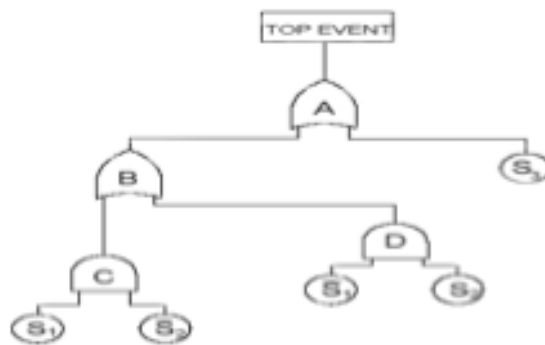
mentre la determinazione dei cut set avviene con il seguente metodo :

- ❖ per la porta AND il primo input della porta AND che si sta risolvendo sostituisce l'indice della porta nella matrice mentre gli altri input sono inseriti nelle successive colonne libere, uno per colonna, sulla stessa riga ove era presente l'indice della porta;
- ❖ per la porta OR il primo input della porta OR sostituisce l'indice che identifica la porta nella matrice e tutti gli altri input sono inseriti nelle successive righe libere uno per riga. Se sono presenti altri elementi nella riga in cui era presente la porta OR questi elementi devono essere ripetuti in tutte le righe in cui sono inseriti gli input della porta OR.

Con riferimento al nostro albero sintetizzato la determinazione dei *Cut-Set* nonché dei *Minimal Cut-Set* è la seguente riportata in figura 6.

Fig. 6 - Determinazione dei *Cut-Set* e *Minimal Cut-Set*.

$$\begin{aligned}
 X_3 &= S_1 \cap \bar{S}_2 \cup \bar{S}_1 \cap S_2 \cup S_3 \\
 &= S_1 \otimes \bar{S}_2 \oplus \bar{S}_1 \otimes S_2 \oplus S_3 \\
 &= S_1 \cdot \bar{S}_2 + \bar{S}_1 \cdot S_2 + S_3
 \end{aligned}$$



A	AB	BC	C	S ₁ - S ₂	S ₁ - S ₂	<div style="border: 1px solid black; padding: 5px; display: inline-block;"> Cutset I Cutset II Cutset III </div>	<div style="border: 1px solid black; padding: 5px; display: inline-block; text-align: left;"> MINIMAL CUT SET </div>
	S ₁	S ₁	S ₁	S ₁	S ₁		
		S ₂		S ₂	S ₂		
			D	S ₁	S ₂		

4. Approcci per gli eventi di base

Al fine di valutare l'affidabilità del sistema complesso si preferisce adottare il modello probabilistico degli eventi (nodi) di guasto elementari, in alternativa, nei casi di difficile soluzione causati dalla insufficienza dei dati sui tassi di guasto dei componenti è possibile utilizzare un approccio "approssimato" con l'utilizzo della logica fuzzy.

La FTA usando il concetto di fuzzy sets è stata studiata solo da alcuni ricercatori in particolare:

- Singer [22,23] utilizza la FTA trattando le frequenze relative degli eventi di base come numeri fuzzy del tipo LR e presenta come esempio l'analisi di un reattore di polimerizzazione e di un centro di fresatura;
- Misra e Weber [19] usano gli operatori max e min per valutare la possibilità del top event in termini di possibilità degli eventi di base ed illustrano la procedura con l'analisi di un sistema di raffreddamento di emergenza di un reattore nucleare;
- Una *event tree analysis* basata sull'uso di affermazioni verbali per caratterizzare la probabilità degli eventi è presentata da Kenarangi [18] che analizza un sistema di protezione per la produzione di energia elettrica;
- Gmytrasiewicz [17] propone un metodo per diagnosticare il guasto dei componenti dall'osservazione di sintomi fuzzy che applica ad un impianto nucleare;
- Sawyer e Rao [20,21] presentano una FTA per l'analisi di sistemi meccanici applicata ad un sistema di valvole in un motore. La tecnica proposta usa numeri fuzzy per determinare i tassi di guasto ed applica l'aritmetica fuzzy per determinare la probabilità del top event e la corrispondente affidabilità del sistema come numeri fuzzy;

- Benedetti [16] studia la possibilità di applicazione della logica fuzzy allo studio dei sistemi industriali, implementando un programma che permette di risolvere alberi di guasto in ambiente fuzzy passando attraverso la determinazione dei *Cut-Sets*.

5. Conclusioni

In sintesi, la combinazione di strumenti proposta per l'analisi di criticità di un sistema informativo di remote banking consente una progettazione accurata dei componenti più a rischio ed una ottimale definizione dell'architettura complessiva del sistema, tenendo conto delle probabilità degli eventi di base e delle loro composizioni mediante operatori booleani. La sintesi finale dell'albero viene effettuata individuando i minimal cut-sets corrispondenti ai componenti critici di guasto.

Maggiore sforzo di indagine può essere profuso verso una modellizzazione più "rozza" della struttura probabilistica degli eventi facendo uso di un approccio fuzzy qualora le probabilità stesse non siano correttamente individuabili.

Bibliografia

- [1] Allen T.J. e Scott Morton M. S., *Information Technology and the Corporation of the 1990s*, New York, (1994).
- [2] Angehrn A., (1997), "The ICDT Model: Towards a Taxonomy of Internet-Related Business Strategies", *European Management Journal*, September.

- [3] Antony R.N., *Sistemi di pianificazione e controllo. Schemi di analisi*, Etas Libri, Milano, (1967).
- [4] Armstrong and Hagel, (1997), *Netgain*, Harvard Business School Press, New York.
- [5] Bazovsky I., *Principi e metodi dell'affidabilità*, Etas Kompass, (1969).
- [6] Booz, Allen and Hamilton, *Internet Banking in Europe: a Survey of Current Use and Future Prospects*, London, (July 1996).
- [7] Di Pillo L., "Phone banking – Le nuove tecniche e quelle trazionali", *Il Sole 24 Ore - Informatica*, (1998).
- [8] Ecchia B., *I processi distributivi nelle imprese bancarie. Nuove tecnologie e strategie di multicanalità*, Edizioni Scientifiche Italiane, (2002).
- [9] ESA European Space Administration, "Fault Tree Analysis in Galileo Projekt", www.galileo-pgm.org.
- [10] Filotto U., *La nuova Web bank. Cultura, organizzazione e tecnologia dopo la crisi di Internet Bancaria*, Editrice Edibank, (2002).
- [11] Lew A. and Tamanaha D. (1976), "Decision table programming and reliability", Proceedings of the 2nd international conference on Software engineering", San Francisco, California, USA, pp. 345-349, .
- [12] Montani S., Portinale L. e Bobbio A., "Dynamic Bayesian Networks for Modelling Advanced Fault Tree Features in Dependability Analysis", TR-INF-2004-03-04-UNIPMN, <http://www.di.unipmn.it/Tecnical-R/TR-INF-2002/index.htm>.
- [13] Naccarato G., *Java database e programmazione client/server*, Apogeo, (2002).

- [14] Romanò e Carignani, *Sistemi informativi per la banca, Cusl*, (2002).
- [15] Salvatici A., *La Banca Virtuale*, Banche e Banchieri, (1996).
- [16] Benedetti P., *Analisi e miglioramento dell'affidabilità di sistemi complessi in ambiente fuzzy*, Tesi di Laurea, Facoltà di Ingegneria Meccanica, Università di Firenze, pp. 169, (A.A. 1993/94).
- [17] Gmytrasiewicz P., Hassberger J.A. e Lee J. C., *Fault tree based diagnostics using fuzzy logic*, IEEE Transactions on pattern analysis and machine intelligence, v.12, n. 11, pp. 1115-1119, (1990).
- [18] Kenarangui R., *Event-tree analysis by fuzzy probability*, IEEE Transactions on reliability, v.40, n.1, pp. 120-124, (1991).
- [19] Misra K. B. e Weber G. G. *A new method for fuzzy fault tree analysis*, Microelectronics and reliability, v. 29, n. 2, pp. 195-216, (1989).
- [20] Rao S. S., *Reliability-based design*, pp. 569, McGraw-Hill, New York, (1992).
- [21] Sawyer J. P. e Rao S. S., *Fault tree analysis of fuzzy mechanical systems*, Microelectronics and reliability, (1994).
- [22] Singer D., *Fault tree analysis based on fuzzy logic*, Computer & chemical engineering, v.14, n. 3, pp. 259-266, (1990).
- [23] Singer D., *A fuzzy sets approach to fault tree and reliability analysis*, Fuzzy sets and systems, v. 34, n. 2, pp. 145-155, (1990).