

Samenvatting

Het doel van dit proefschrift is het generaliseren de methode *masking* die wordt gebruikt ter verbetering van de betrouwbaarheid van digitale systemen. Tevens wordt een methode gepresenteerd die het correct functioneren van een fouten-tolererend systeem onafhankelijk maakt van een onbetrouwbare omgeving. Voor dit laatste doel is een nieuw en efficiënt algoritme voor interactieve consistentie ontwikkeld.

Tot nu toe werden fouten-tolererende digitale systemen veelal beschreven door te verklaren hoe in een bepaalde architectuur de extra, d.w.z redundante, onderdelen of deelsystemen worden benut om de betrouwbaarheid van het totale systeem te verbeteren.

In deze context dient betrouwbaarheidsverbetering te worden beschouwd als de verhouding tussen de Mean-Time-Between-Failures van een fouten-tolererend systeem en de Mean-Time-Between-Failures van een niet-fouten-tolererend systeem met dezelfde functionaliteit.

Fouten-tolererende architecturen worden tegenwoordig met succes toegepast in telefoon centrales, in computers voor het betaalverkeer, ruimtevaart, en zelfs in de burgerluchtvaart.

Het nadeel van de huidige ontwerpmethoden voor fouten-tolererende digitale systemen is het feit dat de betrouwbaarheid niet alleen afhankelijk is van de verbetering die verkregen wordt door het toepassen van een basisarchitectuur, maar dat de betrouwbaarheidsverbetering tevens afhankelijk is van allerlei ontwerpdetails. Bovendien wordt de betrouwbaarheidsverbetering bepaald door de vraag of het architectuurconcept consequent is toegepast of dat bepaalde aanpassingen zijn gemaakt teneinde de kosten te verlagen. Voor veel ontwerpen van fouten-tolererende digitale systemen blijkt het erg moeilijk te zijn om gedurende het ontwerpproces de ontwerpbeslissingen te

herkennen die kritisch zijn voor de betrouwbaarheidsverbetering. In feite komt het er op neer dat als gevolg van de toegepaste ontwerpmethodes de betrouwbaarheidseigenschappen niet kunnen worden geverifieerd tijdens het ontwerpproces. Dus de uiteindelijke betrouwbaarheidsverbetering van het fouten-tolererend systeem hangt sterk af van de kwaliteit van het ontwerp, het ontwerpproces en het validatieproces. In ieder geval is de berekende of geschatte betrouwbaarheidsverbetering van veel ontwerpen twijfelachtig.

In dit proefschrift wordt beschreven hoe voor de klasse van fouten-tolererende digitale systemen die gebaseerd zijn op “masking” een aantal van deze nadelen weg te nemen zijn door het fouten-tolererend systeem te reduceren tot een verzameling gekoppelde Moore machines en door de kritische data communicaties in een dergelijk systeem te identificeren.

Bovendien zal het klassieke masking concept worden gegeneraliseerd door de meerderheidsfunctie, die de kern vormt van deze methode, te vervangen door een decodeerfunctie van een foutencorrigerende code. Het resultaat hiervan zullen we “Generalized Masking” noemen.

De totale klasse van fouten-tolererende digitale systemen, die gebaseerd is op Generalized Masking met inbegrip van de klassieke masking-systemen kan, indien de systemen op een voldoende hoog nivo van abstractie beschreven zijn, door twee deelklassen worden gekarakteriseerd, afhankelijk van de wijze waarop de deelsystemen verbonden zijn. Deze twee deelklassen worden gekarakteriseerd door respectievelijk twee functies, nl. \mathcal{X} en \mathcal{Y} of door drie functies nl. \mathcal{X} , \mathcal{Y} en \mathcal{Z} , alsmede door het maximum aantal T te tolereren defecte modules. De systemen die door deze twee klassen beschreven worden zijn respectievelijk $(\mathcal{X}, \mathcal{Y}, T)$ systemen en $(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, T)$ systemen.

De functie \mathcal{X} beschrijft de manier waarop ieder van de modules in het fouten-tolererend systeem zijn informatie ontvangt van de buitenwereld. We zullen aantonen dat deze functie altijd correct moet worden uitgevoerd ook wanneer die buitenwereld foutieve en misleidende informatie naar het systeem stuurt. Is aan deze voorwaarde niet voldaan, dan kan het fouten-tolererend systeem zich incorrect gaan gedragen zelfs wanneer minder modules defect zijn dan volgens het ontwerp is toegestaan. Dit wordt het “Input Problem” genoemd.

Het correct uitvoeren van de functie \mathcal{X} houdt in dat de correct functionerende modules in het fouten-tolererend systeem allemaal tot dezelfde conclusie

moeten komen betreffende de informatie die zij vanuit de externe bron hebben ontvangen. Wanneer die externe bron correct functioneert dan moet die conclusie overeenkomen met de data die door die externe bron was verstuurd.

Algoritmen die soortgelijke eigenschappen bezitten als de eigenschappen die vereist zijn voor de functie \mathcal{X} zijn de zogenaamde algoritmen voor interactieve consistentie of Byzantijnse Generaals Algoritmen. Sinds 1978 worden deze algoritmen onderzocht en sindsdien zijn vele resultaten gepubliceerd.

De algoritmen voor Interactieve Consistentie hebben het nadeel dat wanneer twee of meer fouten getolereerd moeten worden, een enorme hoeveelheid data tussen de modules van het fouten-tolererend systeem moet worden uitgewisseld. Voor praktische toepassingen betekent dit dat niet meer dan drie of vier defecte modules in een systeem getolereerd kunnen worden. Teneinde de hoeveelheid data die verzonden moet worden te verminderen, wordt een nieuwe klasse van algoritmen voor interactieve consistentie gepresenteerd, die minder communicatie vereist indien het aantal te tolereren defecte modules kleiner is dan vier. Helaas wordt geen verbetering verkregen voor het meest eenvoudige algoritme, nl. het algoritme dat geschikt is voor vier modules waarvan er ten hoogste één fout mag zijn.

Tenslotte worden in dit proefschrift een aantal methoden gepresenteerd die het Input Problem oplossen. Deze algoritmen zijn gebaseerd op algoritmen voor interactieve consistentie.

Samenvattend is het doel van dit proefschrift

- De generalisatie van de ver- N -voudiging methode, welke gebaseerd is op een gedistribueerde uitvoering van een foutencorrigerende code. Het resultaat noemen we “Generalized Masking”.
- De definitie van twee klassen fouten-tolererende systemen die de klasse van systemen die gebaseerd zijn op Generalized Masking karakteriseren.
- De presentatie van een bepaalde architectuur, het (N, K) -concept die gebaseerd is op Generalized Masking en waarvan de voordelen in de praktijk zijn bewezen.

- De presentatie van een symboolcorrigerende code ten behoeve van het $(4, 2)$ concept, die naast de symboolfouten ook nog in staat is bit-fouten te corrigeren zonder dat daar extra redundantie voor nodig is.
- De definitie van het Input Problem van een fouten-tolererend systeem.
- De presentatie van een nieuwe klasse van synchrone deterministische algoritmen voor interactieve consistentie die gebaseerd is op meerderheidsbeslissingen en foutencorrigerende codes en die in praktische toepassingen minder data communicatie vereist dan de bestaande synchrone deterministische algoritmen voor interactieve consistentie.
- De oplossing van het Input Problem met behulp van gelijksoortige algoritmen als de algoritmen voor interactieve consistentie.

Summary

This thesis attempts to generalize a particular method, called *masking*, which is used for improving the reliability of digital systems, such as computer systems. Moreover a method is presented which makes the proper functioning of a fault-tolerant system independent of an unreliable external world. For the latter purpose a new and effective interactive consistency algorithm is developed.

Thus far fault-tolerant digital systems are mostly described just by explaining how the spare (i.e. redundant) components or subsystems in a particular architecture are utilized for improving the overall system reliability.

In the present context the reliability improvement should be interpreted as the ratio between the mean time between failures of the fault-tolerant system and the mean time between failures of a non-fault-tolerant system with the same functionality.

At present many fault-tolerant architectures are successfully applied in real world systems, such as telephone exchanges, space vehicles, computers for transaction processing, etc., and even in civil aviation.

The drawback of the current design methods for fault-tolerant systems however is that reliability improvement not only depends on the improvement achieved by the application of some basic architecture, but that the reliability also depends on many design details and whether the architectural ideas are implemented straightforwardly or whether some adaptations have been made in order to arrive at a more cost-effective design. In many fault-tolerant designs, design decisions which are critical with respect to the reliability improvement, are very difficult to recognize during the design process. In fact it often turns out that due to the design methods applied, the reliability properties of the system cannot be verified during the design process. Hence the

quality of the design, the design process and the validation process heavily determines the final reliability of the fault-tolerant system. At least the calculated or estimated reliability improvement of many of the present designs is questionable.

In this thesis we will try to overcome some of these drawbacks for the class of fault-tolerant architectures which are based on masking, by reducing a fault-tolerant digital system which is based on masking to a set of coupled Moore machines and identifying the critical data transfers.

Moreover, the classical masking concept, the N -modular redundancy scheme, will be generalized by replacing the majority vote, which is the key of this method, by the decoder function of an error-correcting code. The result will be called "Generalized masking".

Provided the systems are described on a sufficiently high level of abstraction, the entire class of fault-tolerant systems based on generalized masking, the classical masking systems inclusive, can be described in two ways depending on the interconnection of the subsystem. One of these subclasses is characterized by two functions, i.e. \mathcal{X} and \mathcal{Y} and the number T of faulty modules that can be tolerated. The other subclass is characterized by three functions \mathcal{X} , \mathcal{Y} and \mathcal{Z} and the number T of faulty modules that can be tolerated. The systems described by these two classes are called $(\mathcal{X}, \mathcal{Y}, T)$ systems and $(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, T)$ systems respectively.

The function \mathcal{X} describes the way in which each of the modules of the fault-tolerant system receives its information from the outside world. We will show that this function must be performed "fault free" even if the outside world produces incorrect data, otherwise the fault-tolerant system might go down even if it contains less faulty modules than it is designed to tolerate. This will be called the "Input Problem".

A "fault free" performance of the function \mathcal{X} means that the correctly functioning modules in the fault-tolerant system all must come to the same conclusion about what the external source has sent them. And if the external source functions correctly, this conclusion should be the data which were sent by the external source.

Algorithms with properties similar to those which are required for the function \mathcal{X} are the so-called Interactive Consistency Algorithms or Byzantine

Generals Algorithms. They have been investigated since 1978 and many results have been published since then.

The interactive consistency algorithms suffer from the fact that if two or more faulty modules are to be tolerated an enormous amount of data has to be transmitted between the modules of the fault-tolerant system. In practice this means that no more than three or four faulty modules can be tolerated in a system. In order to reduce the amount of data which has to be transmitted, a new class of interactive consistency algorithms will be presented which is based on error correcting codes and which if the number of faulty modules is four or less, requires less data to be transmitted than the existing algorithms. Unfortunately no improvement is obtained for the most simple algorithm which runs on 4 modules of which at most one may be faulty.

Finally we will present a number of methods which solve the Input Problem. These methods are based on an algorithm similar to the interactive consistency algorithms.

In summary this thesis aims at

- A generalization of the N -modular redundancy scheme which is based on the distributed implementation of error-correcting codes, and which will be called generalized masking.
- A definition of the two classes of systems which characterize the systems that are based on generalized masking.
- The presentation of a particular architecture, called the (N, K) -concept, which is based on generalized masking and the feasibility of which is proved by application in a commercial system.
- The presentation of a symbol-error-correcting code to be used in the $(4, 2)$ -concept, which in addition to symbol-errors is also capable of correcting bit errors without requiring extra redundancy.
- A definition of the Input Problem of a fault-tolerant system.
- The presentation of a new class of interactive consistency algorithms which is based on voting and coding, and which requires in most practical applications less data transfer than the existing synchronous deterministic interactive consistency algorithms.

- The solution of the Input Problem on the basis of algorithms similar to the interactive consistency algorithms.

Preface

The field of fault-tolerant computing is still rather new. This can be concluded from a still continuing discussion on definitions and a lack of standard literature in which the area is treated from a formal point of view instead of from a phenomenological point of view. Therefore a short introduction to the field of fault-tolerant computing and one of its most intriguing issues, the so-called “Input problem” is presented in Chapter 1. In this chapter subsequently the aspects which determine the reliability of a digital system are discussed, the relevant reliability criteria are defined, and a survey is given of the various methods and techniques which are available for improving the reliability of digital systems such as fault avoidance and fault-tolerance based on error detection, masking redundancy or dynamic redundancy.

Futhermore, in this chapter we will point out that the method to be used often depends on the required form of reliability (fail-safe, fault-tolerant, survivable without repair) and the degree of improvement to be achieved.

In a separate section the arguments are presented which support the opinion that repairable fault-tolerant systems should be implemented by means of masking redundancy, the latter being the subject of this thesis.

Finally the “Input problem” which is an integral part of any fault-tolerant system will be explained.

In Chapter 2 the well known N -modular redundancy scheme will be generalized to a class of systems which we will call “Generalized masking redundancy”. As an introduction to Generalized masking first a particular architecture, called the (N, K) -concept, which belongs to this class will be presented. This new fault-tolerant computer architecture is based on a “distributed implementation” of a symbol-error-correcting code. The faults in this (N, K) -concept are masked by this error-correcting code instead of by a majority vote function which is the case in N -modular redundant systems.

To understand better the time dependency of a synchronous digital system we will model a synchronous digital system by means of the Moore model and relate time and space by unfolding time into space.

Using as a basis the unfolded representation of the N -modular redundancy scheme we will identify and discuss the critical data transfers. We will show that the broadcast of data and the voting on the results can be replaced by the encoder function and the decoder function of an error-correcting code respectively. This can be implemented for the I/O of the system as well as for the state of the system. This results in the definition of a $(\mathcal{X}, \mathcal{Y}, T)$ fault-tolerant system and a $(\mathcal{X}, \mathcal{Y}, \mathcal{Z}, T)$ fault-tolerant system. Real fault-tolerant systems based on generalized masking will be based on a mixture of both. The basic ideas behind Generalized masking could also be described in terms of a “distributed implementation of an error-correcting code” or in terms of “the encoding of physically implemented functions”.

The (N, K) -concept is described in detail for $N = 4$ and $K = 2$.

It will be shown that symbol-error-correcting codes with additional bit-error-correcting capabilities make additional memory protection by means of bit-error-correcting codes superfluous and a newly designed symbol- and bit-error-correcting code for the $(4, 2)$ -concept will be presented.

The systems described in Chapter 2 are all based on the assumption that the Input Problem is solved. In Chapter 5 this finally will be done on the basis of interactive consistency algorithms. Chapters 3 and 4 will be devoted to these interactive consistency algorithms.

In Chapter 3 the Byzantine Generals problem, which is also called the Interactive consistency problem, will be sketched based on its original description. Its relevant parameters will be discussed and the requirements which have to be fulfilled by an algorithm which solves the problem are defined. Thereafter a survey will be given of the existing literature and the results obtained so far.

In the second part of Chapter 3 a new class of algorithms will be defined which will be called Dispersed Joined Communication algorithms and which satisfy some properties which can be regarded as a more liberal version of the interactive consistency requirements.

Based on these Dispersed Joined Communication algorithms a new class of

algorithms for reaching interactive consistency will be presented. This class of algorithms is based on voting and error-correcting codes and meets both the $N \geq 3T + 1$ bound and the $K \geq T + 1$ bound.

The class of Interactive Consistency algorithms based on voting and error-correcting codes comprises:

- the class of algorithms based on voting published in the early eighties, which we will call the classical algorithms.
- a new class of algorithms based on voting which require considerably less data communication than the classical algorithms and which meet both the $K \geq T + 1$ bound and the $N \geq 3T + 1$ bound.

The class of algorithms described in Chapter 3 contains algorithms which require much less data communication between the modules than the existing synchronous deterministic algorithms. In order to compare the new algorithms defined in Chapter 3 with the existing synchronous deterministic algorithms two criteria will be defined, i.e.:

- the number of messages which needs to be transmitted between the modules,
- the minimum size of the original message.

For these criteria a number of relations will be derived which make it possible to calculate these figures. For a large number of practical examples the resulting figures are presented.

Although the number of messages in our new class of algorithms based on voting and error-correcting codes, increases exponentially with the number of faults which are to be tolerated and the number of messages in one of the algorithms published by Dolev grows polynomial with the number of faults which are to be tolerated, we will show that for practical applications the algorithms in the class of algorithms which is based on voting and coding are favourable.

In Chapter 5 a method is presented which makes the proper functioning of a fault-tolerant system independent of an unreliable external world. In other words, the solution to the Input Problem will be presented. This solution will be extended to a general solution for the interconnection of fault-tolerant systems.

The correctness of the behaviour of a fault-tolerant system depends among other things on the correct distribution of the data of unreliable I/O devices over the modules of the fault-tolerant system. A malfunctioning system, whether it is fault-tolerant or not, should never defeat a correctly functioning fault-tolerant system, i.e a system which does not contain more faulty modules than it is designed to tolerate. In order to cope with this problem, in Chapter 5 interactive consistency of communicating fault-tolerant systems will be defined. Thereafter a number of interconnection methods and algorithms will be presented which satisfy the above-mentioned interactive consistency. These interconnection methods and algorithms are all based on interactive consistency algorithms. The implementation of such an algorithm for interactive consistency between communicating fault-tolerant systems is described in detail for the (4,2)-concept fault-tolerant computer system architecture.