Informatik-Fachberichte 121

Herausgegeben von W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Klaus Echtle

Fehlermaskierung durch verteilte Systeme

Springer-Verlag
Berlin Heidelberg New York Tokyo

Autor

Klaus Echtle
Institut für Informatik IV, Universität Karlsruhe
Zirkel 2, 7500 Karlsruhe 1

CR Subject Classifications (1985): C.2.2, C.2.4, D.4.5

ISBN-13: 978-3-540-16464-7 e-ISBN-13: 978-3-540-16464-7
DOI: 10.1007/978-3-540-16464-7

2145/3140–543210

Kurzfassung

Bestehen in Mehrrechnersystemen gleichzeitig hohe Zuverlässigkeits- und Reaktionsgeschwindigkeits-Anforderungen, so sind häufig Fehlertoleranz-Verfahren zu wählen, die auf statischer Redundanz (bzw. Hybridredundanz) beruhen, etwa in Form von 2-von-3-Prozeßsystemen. Dem Vorteil der einfachen Diagnose und schnellen Maskierung von Einzelfehlern beliebiger Art steht als Nachteil ein hoher Redundanzaufwand gegenüber, bedingt durch die Verdreifachung der Prozesse sowie der zugeordneten Rechner. Erfolgt die Fehlermaskierung bei der Interprozeßkommunikation, so sind Nachrichten zwischen allen drei redundanten Exemplaren von Sender- und Empfängerprozeß zu transferieren, wodurch die Belastung des Kommunikationssystems sogar auf das Neunfache steigt. Dies führt i.a. zur Entwicklung von redundanz-angepaßten Verbindungsstrukturen oder leistungsfähigerer Kommunikationssysteme.

Im Gegensatz dazu wird in dieser Arbeit ein Fehlertoleranz-Verfahren vorgeschlagen, das statische Redundanz in der Verarbeitung mit dynamischer Redundanz bei der Kommunikation kombiniert und so die Anzahl der zu transferierenden Interprozeßnachrichten auf etwa 1/3 senkt (oder konfigurationsbedingt auf eine noch niedrigere Anzahl). An die Stelle der voneinander unabhängigen Instanzen zur 2-von-3-Mehrheitsentscheidung bei jedem Empfängerexemplar (d.h. bei jedem der drei redundanten Exemplare des Empfängerprozesses) tritt eine einzige Instanz, die bei den Sendern angeordnet ist und dort den Übergang zur dynamischen Redundanz ermöglicht. Um selbst fehlertolerant zu sein, wird die Fehlermaskierungs-Instanz als verteiltes System (bezogen auf das Gesamtsystem: als verteiltes Subsystem) implementiert - bestehend aus drei sogenannten Maskierungs-Knoten, die je einem Senderexemplar zugeordnet sind und miteinander kooperieren, um eine 2-von-3-Mehrheitsentscheidung zu treffen. Der zusätzlich entstehende Interaktionsaufwand der drei Maskierungs-Knoten bleibt gering, wenn effiziente Signaturbildungs-Verfahren zur Verringerung der Nachrichtenlänge und Transferprioritäten zur Bevorzugung von Kurznachrichten existieren.

Das senderseitig angeordnete verteilte System zur Fehlermaskierung kann den Kommunikationsaufwand weiter senken, indem es für jeden Empfänger einer Interprozeßnachricht unter den fehlerfreien Senderexemplaren das auswählt, das voraussichtlich den geringsten Kommunikationsaufwand verursacht, z.B. aufgrund einer Nachbarschaft zwischen Sender und Empfänger. Durch diese Senderauswahl entsteht eine Wechselwirkung mit dem Re-/Konfigurator, der nicht mehr alle, sondern nur noch je eines der redundanten Exemplare von häufig miteinander kommunizierenden Prozessen möglichst eng benachbarten Rechnern zuordnen muß.

Das Konzept der Fehlermaskierung durch verteilte Systeme wird durch einen Maskierungs-Protokoll-Graphen formalisiert, dessen Semantik aus einem geeignet attributierten Petri-Netz abgeleitet ist. Ein Fehlertoleranz-Kriterium gestattet, die Korrektheit eines in Form dieses Graphen gegebenen Protokolls zu überprüfen und damit das Feld aller Implementierungs-Möglichkeiten von verteilten Systemen zur Fehlermaskierung abzugrenzen.

Eine programmiersprachliche Implementierung der Maskierungs-Knoten konkretisiert eines der zulässigen Protokolle. Da diesem nur schwache Annahmen über Hardware, Betriebs- und Kommunikationssystem zugrundeliegen, dürfte sich das Verfahren für Systeme aus Standardkomponenten eignen, was jedoch seine gute Kombinierbarkeit mit speziellen Einrichtungen (z.B. zum 1:x-Transfer, engl. multicasting) nicht ausschließt. Eine Protokoll-Verifikation und eine durch Simulation gewonnene quantitative Bewertung ergänzen diesen Teil der Arbeit.

Abschließend richtet sich die Betrachtung auf einige Grenzfälle der Fehlermaskierung durch verteilte Systeme. Ein spezielles Protokoll führt zu einer weiteren Senkung des redundanten Kommunikationsaufwands - jedoch um den Preis einer erhöhten Ausführungsdauer. Außerdem wird der Übergang zu n-von-m-Prozeßsystemen ($m \geq 3$) zur Tolerierung von symptom-gleichen und symptom-verschiedenen Mehrfachfehlern vollzogen.

Stichworte

Fehlertoleranz, statische Redundanz, Hybridredundanz, Fehlermaskierung, 2-von-3-System, TMR, n-von-m-System, NMR, verteiltes System, Protokoll, Kommunikation, Verifikation, Simulation, Modellierung von Fehlertoleranz-Verfahren, Petri-Netz.

Abstract

Fault-Masking by Distributed Systems

High reliability and short response times can be achieved by fault-tolerant systems, using static redundancy, for example 2-out-of-3-systems (TMR). Fault masking with inter-process communication requires messages between all redundant sender and receiver processes and thus causes a ninefold communication expense. This thesis proposes a sender-located voter and the combination of static redundant processing with dynamic redundant communication. Resulting advantages are the decrease of the number of inter-process messages to about one third (threefold communication expense) or less, as well as the selection of the most suitable links, before inter-process messages are transferred. As the failure status couldn't be determined at a single sender node, in particular not at the faulty one, the voter is realized as a distributed (sub-) system, consisting of three nodes. Its protocol represents a software implementation of fault-tolerance, which is nearly independent of the underlying hardware and communication structure. The set of correct voter-protocols is specified by a so-called voter-protocol graph, whose semantics is explained by Petri nets and corresponding attribute-transformation rules. One of the protocols is assessed quantitatively by simulation.

Keywords

Fault-Tolerance, Static Redundancy, Hybrid Redundancy, Fault-Masking, 2-out-of-3-System, TMR, n-out-of-m-System, NMR, Distributed System, Protocol, Communication, Verification, Simulation, Modelling of Fault-Tolerant Systems, Petri Nets.

Inhaltsverzeichnis