Abstract In the paper a formal model of the fault-tolerant distributed computer system with the possibility of reconfiguration is considered. A fault-tolerant distributed computer system model from hardware viewpoint forms a fault-tolerant net, in which concurrent algorithms are performed. Due to uniform criteria of reconfigurations and interactions, an uniform analysis, synthesis and optimization is possible. Next the main idea of the operating system for the above systems has been described.

1. Introduction

The idea of fault-tolerant computing has appeared at the same time as unreliable computing devices.

To achieve fault-tolerance, efficient procedures must be implemented which identify and isolate faulty processing units (device, chip, gate, program, data, etc.) within a reasonable time after faults occurrence. The next step would be connected with repairing the faulty part of the system. In the above situation self-repairing of the system would be desired. Self-correction of software parts of the computer system and a replacement of faulty hardware by redundant units are done in practice. As it has been formally proved by von Neumann [17] also the self-repairing (self-reproducing) of hardware parts of computer systems is possible. Actually constructed systems are not in the position to do it. However, they have built-in mechanisms which preserve correct work in spite of errors - in other words they constitute the fault-tolerance. This property is received by following procedures:

- fault detection (hardware and software mechanism used to determine if a fault occurs),
- fault containment (techniques used to prevent fault-damaged information from propagating through a system),
- fault diagnosis (hardware and software techniques used to locate and identify a fault),
- fault correction (self-reparation achieved by means of software
and hardware redundancy).

This paper deals with the problem of fault-correction limited to self-reconfiguration. In the paper the main idea of the operating system for fault-tolerant distributed computer systems (abbr. FTDCS) has been described.

Formally, our approach to FTDCS is based on the notion of the algebraic model of the distributed computer system (Just [12]) and on the theory of selfmodifiable algorithms (Eberbach [6]). In [8] a formal model of FTDCS has been described. In [13] we have restricted our attention to the process of synthesis, i.e. allocation of actions to particular processing units so that the system works independently of processor faults. Control in FTDCS has been presented in [9]. This paper is a continuation of [8,9,13].

FTDCS model from hardware viewpoint forms a fault-tolerant net, in which concurrent algorithms (a concurrent algorithm system) with changeable interaction are performed. The behaviour of a concurrent algorithm system is described by means of a canonical set of equations. FTDCS is a system with two tolerance relations defined on sets of processing units and algorithms. These tolerances express possibilities of FTDCS behaviour changes, caused by faults, interrupts, requests of resources, etc.

Next the uniform criteria of possible directions of reconfiguration and interaction are considered, namely cost functions of expectation, replacing and interaction. A cost function of actions and introducing a cost system with "beyond-real" numbers make possible estimating costs of algorithms' performance.

In paper 4 problems of an optimization have been described, i.e. totally, CAS, partially and one-step ahead optimal problems.

Next, on the basis of the received results some modules of the FTDCS operating system have been outlined.

The concepts, introduced in our paper are thought as formal tools of designing the wide class of fault-tolerant networks of computers.

2. The model

By the fault-tolerant distributed computer system (abbr. FTDCS) we mean any system:

\[ \text{FTDCS} = (\text{FTN}, \text{CAS}, h_0) \]

where:

- \( \text{FTN} \) - a fault-tolerant net,
- \( \text{CAS} \) - a concurrent algorithm system,
- \( h_0 \) - an initial allocation relation.