Reliability of Cluster System with a Lot of Software Instances

Magdalena Szymczyk and Piotr Szymczyk

Department of Control Systems, University of Mining and Metallurgy, al. Mickiewicza 30
30-059 Kraków, Poland
{mszm, pszm}@ia.agh.edu.pl

Abstract. This paper presents model of complex fault – tolerant system with multiple software instances and hardware clusters. We want to show influence of number software and hardware components on overall reliability of the system. Previously, other models have been developed only for software or hardware systems. Our model assumes that failure of each component is statistically independent.

1 Introduction

Today computers are much more sophisticated then their earlier version. They are made up of software and hardware components.

These days, in order to ensure effective performance of the computer, software and hardware need to function with considerable amount of reliability. There are number of methods to improve reliability of software products: reliable software design, fault-tolerance design, formal methods and testing [1] [2] [3].

Reliability is the probability that system works properly by a certain time.
Reliability of serial systems where failures of components in system are mutually independent equals with the probability that all components are working [4] [5].

If the probability that a component A_i is operational equals $P(A_i)$, and its reliability equals $R_i=P(A_i)$, then reliability of the whole system (probability that system is functioning properly) with the assumption of independency of failures equals:

$$R_s = \prod_{i-1}^{n} R_i \tag{1}$$

For parallel system reliability R_p equals to:

$$R_p = 1 - \prod_{i=1}^{n} (1 - R_i) \tag{2}$$

under the assumption of independence of failure events.

M. Bubak et al. (Eds.): ICCS 2004, LNCS 3036, pp. 417–420, 2004.

2 Optimization of the Number of Software and Hardware

Let's make some assumption in our model. A repeated task and each repetition of task are independent. We assume that software task has a fixed probability of failure. Software failures of different version of code are independent, but still have the same of reliability. For this architecture we have such RBD diagram [5]:

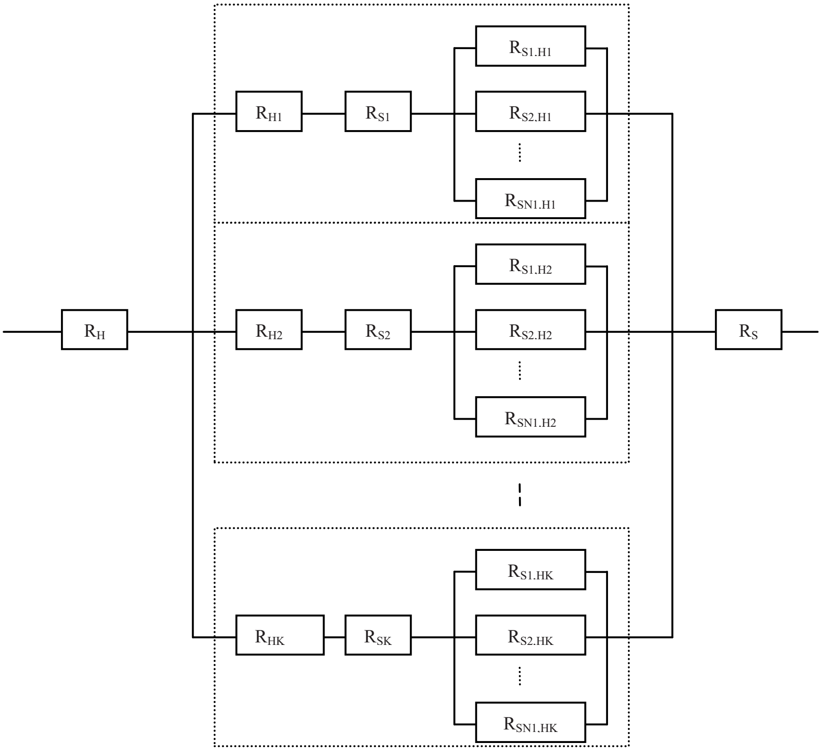

Fig. 1. RBD diagram for considered system

Notation:

K – number of hardware clusters in systems
N – number of software instances on j-th cluster node
H_j – j-th hardware node
S_i,H_j – i-th software instance on j-th cluster node
R_H – common reliability of hardware
R_{Hj} – reliability of each hardware node
R_{Sj} – reliability of common software on j-th hardware node
$R_{Si,Hj}$ – reliability of each software instance on j-th hardware node
R_S – reliability of common software

Reliability of the system is given by:

$$R = R_H * \left(1 - \prod_{j=1}^{K}\left(1 - R_{H_j} * R_{S_j} * \left(1 - \prod_{i=1}^{N_j}\left(1 - R_{S_i,H_j}\right)\right)\right)\right) * R_S \tag{3}$$

from this equation , if all R_{Hj} are equal, R_{Sj} and $R_{Sj,Hj}$ too, we get:

$$R = R_H * R_S * \left(1 - \left(1 - R_{H_K} * R_{S_N} * \left(1 - \left(1 - R_{S_N,H_K}\right)^N\right)\right)^K\right) \tag{4}$$

Results can be presented after some calculations in a form of charts. For simplicity and better visualization of influence R_{HK} and $R_{SN,HK}$ on R it is assumed that R_H, R_S, and R_{SN} are equal to 1. Values R_H, R_S and R_{SN} in all cases we have a direct influence on R, so it should have the highest value possible.

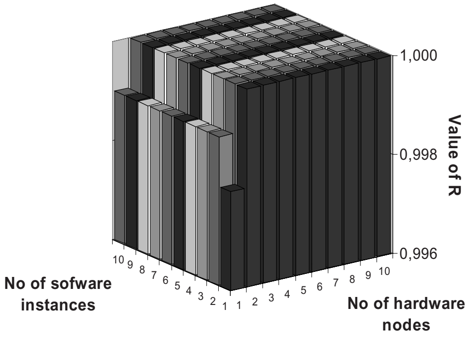

Fig. 2. Charts for R_H=1, R_S=1, R_{HK}=0.999, R_{SN}=1, $R_{SN,HK}$=0.999

From Fig. 2 it follows that for software instances greater than 4 and for the same numbers of nodes the value of R does not change at all. So if we give greater numbers of software and clusters, R doesn't change dramatically. When R_{HK} and $R_{SN,HK}$ are equal, increasing number of hardware nodes to 2 produce good result on R.

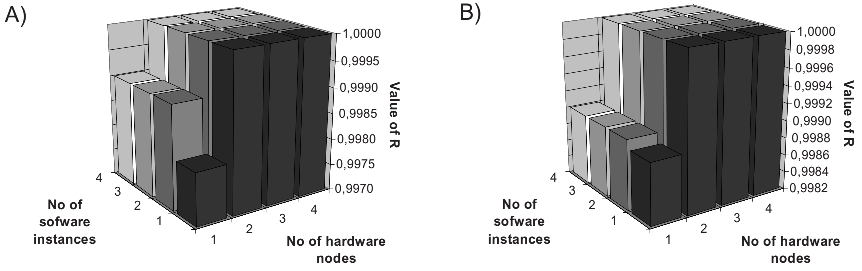

Fig. 3. Charts for R_H=1, R_S=1 R_{SN}=1, R_{HK}=0.999 A) $R_{SN,HK}$=0.999; B) $R_{SN,HK}$=0.9999;

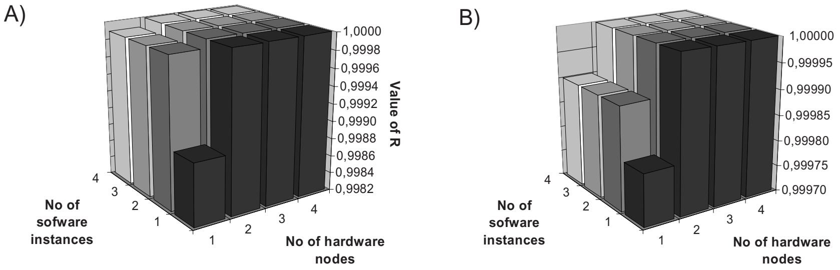

Fig. 4. Charts for R_H=1, R_S=1 R_{SN}=1, R_{HK}=0.9999 A) $R_{SN,HK}$=0.999; B) $R_{SN,HK}$=0.9999;

3 Conclusions

We consider dependencies in the system reliability with respect to the number of hardware cluster nodes and software instances for fault-tolerant systems. Analyzed system is composed of K clusters and N software instances. For such system reliability R is most sensitive on changes from 1 to 2 in hardware nodes in all considered cases. The most optimal number of hardware nodes equals 2 because further increase in hardware does not have any remarkable influence on R. In principle, software depends from properly functioning hardware so if we have very high $R_{SN,HK}$ values with relation to R_{HK}, increase in software number dose not produce the same effects as increase in node number. It can raise reliability R if $R_{SN,HK}$>>R_{HK} (Fig.3 B) and software instances changes from 1 to 2. Such solution can be economically justified in the case when the implementation of two different instances of software is significantly less than the cost of buying two hardware nodes and $R_{SN,HK}$>>R_{HK}. Increasing above two software instances and two hardware nodes don't have significant influence on the system reliability. It is necessary to focus our attention on obtaining high level of values $R_{SN,HK}$ and R_{HK}.

References

1. Kim K.H. and Welch H.O.: Distributed execution of recovery blocks: an approach for uniform treatment of hardware and software faults in real-time applications, IEEE Transaction on Computers, 38 (5): pp. 626-636, 1989.
2. Randell B.: System Structure for software fault tolerance. IEEE Transaction Software Engineering, SE-1(2): pp. 220-232, 1975.
3. Avizienis A.: The N-version approach to fault-tolerant, SE-11(12): pp. 1491-1501, 1985.
4. Hunter S.W., Smith W.E.: Availability Modeling and Analysis of a Two Node Cluster.
5. Bobbio A. and Trivedi K.: Reliability Theory and Methods