

République Algérienne Démocratique Et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique

Université colonel HADJ LAKHDAR –BATNA-
Faculté des sciences de l'ingénieur
Département d'informatique

Mémoire

présenté

pour obtenir le grade de Magistère en informatique

par

Malika Bachir

Spécialité : Système d'Information et de Communication (SIC)

Titre

Tolérance aux fautes des systèmes temps-réel embarqués basée sur la redondance

Encadreur : Dr Zidani Abdelmadjid

Co-Encadreur : Dr Kalla Hamoudi

Composition du jury :

Dr Mohammed Benmohammed (<i>prof.</i>)	Président	<i>Université de Constantine</i>
Dr Belami Azzedine (<i>MCA</i>)	Examineur	<i>Université de Batna</i>
Dr Belattar Brahim (<i>MCA</i>)	Examineur	<i>Université de Batna</i>
Dr Zidani Abdelmadjid (<i>MCA</i>)	Rapporteur	<i>Université de Batna</i>
Dr Kalla Hamoudi (<i>MCB</i>)	Co-Rapporteur	<i>Université de Batna</i>

2010-2011

Table des matières

Liste des figures	v
-------------------------	---

Liste des tableaux	vi
--------------------------	----

Chapitre 1 Introduction générale

1.1.Problématique générale	7
1.2.Apport du mémoire	9
1.2.1Méthodologie AAA-FAULT ^{t.ind}	9
1.2.2Méthodologie AAA-FAULT ^{t.dep}	9
1.3.Plan du mémoire	10

Chapitre 2 Introduction à l'ordonnancement dans les systèmes distribués temps réel embarqués

2.1.Définitions	11
2.1.1.Système réactif et système temps réel.....	11
2.1.2.Système distribué et système embarqué.....	15
2.2.Spécifications des systèmes distribués temps réel embarqués	16
2.2.1.Spécification algorithmique	16
2.2.2.Spécification matérielle.....	17
2.2.3.Contraintes temporelles et d'embarquabilités	18
2.3.Problème de distribution et d'ordonnancement temps réel	19
2.3.1.Terminologies	19
2.3.2.Présentation du problème de distribution/ordonnancement	20
2.4.Classes d'algorithmes de distribution et d'ordonnancement temps réel	20
2.4.1.Algorithmes hors-ligne et enligne	21
2.4.2.Algorithmes exactes et approchés	21
2.5.Algorithme de distribution et d'ordonnancement de SYNDEX	21
2.5.1.Spécification de l'Algorithme de distribution/ordonnancement de SynDEx	22
2.5.2.Présentation de l'algorithme de distribution et d'ordonnancement de SynDEx	23
2.6.Conclusion	26

Chapitre 3 Tolérance aux fautes dans les systèmes distribués temps réel embarqués

3.1.Introduction	27
3.2.Terminologies	28
3.2.1.Degré de permanence des fautes	30
3.2.2.Hypothèses de défaillances (Classes de défaillances).....	31
3.2.3.Techniques de tolérance aux fautes.....	32
3.2.4.Algorithmes de la tolérance aux fautes	33
3.2.5.Problème de distribution et d'ordonnancement temps réel et tolérant aux fautes.....	36
3.3.Conclusion	37

Chapitre 4 Description du problème

4.1.Introduction	38
4.2.Classification des machines	38
4.2.1.Traitement (séquenceur).....	39
4.2.2.Mémoire	40
4.2.3.Communication	40
4.3.Modèle d' architecture	42
4.3.1.Architecture à liaisons point à point.....	45
4.3.2.Architecture à liaisons bus	46
4.4.Modèle d'algorithme	47
4.5.Modèle d'implantation	49
4.6.Conclusion	52

Chapitre 5 Etat de l'Art

5.1.Introduction	53
5.2.Stratégies de réplication	54
5.2.1.Réplication active.....	54
5.2.2.Réplication passive.....	56
5.2.3.Réplication semi-active (hybride)	59
5.3.Conclusion	61

Chapitre 6 Méthodologies proposées pour des architectures à liaison bus

6.1.Introduction	62
6.2.Présentation de l’algorithme de distribution/ordonnement tolérant aux fautes des tâches indépendantes AAA-FAULT^{t.ind}	64
6.2.1.Modèle de fautes	64
6.2.2.Données du problème	64
6.2.3.Principe général de la méthodologie AAA-FAULT ^{t.ind}	65
6.3.Présentation de l’algorithme de distribution/ordonnement tolérant aux fautes des tâches dépendantes AAA-FAULT^{t.dep}	77
6.3.1.Modèle de fautes	77
6.3.2.Données du problème	77
6.3.3.Principe général de la méthodologie AAA-FAULT ^{t.dep}	78
6.4.Prédiction du comportement temps réel	89
6.5.Conclusion	89

Chapitre 7 Evaluation de la méthodologie AAA-FAULT^{t.dep}

7.1.Introduction	90
7.2.Paramètres d’évaluation	90
7.3.Les résultats	91
7.3.1. Effet du rapport entre le temps moyen de communication et le temps moyen d’exécution sur AAA-FAULT ^{t.dep}	91
7.3.2. Effet du nombre de tâches sur AAA-FAULT ^{t.dep}	91
7.3.3. Effet du nombre de processeurs sur AAA-FAULT ^{t.dep}	92
7.4.Générateur de graphe d’algorithme	93
7.5.Conclusion	94

Conclusion et Perspectives	95
---	----

Bibliographie	97
----------------------------	----

Résumé :

Les systèmes distribués temps réel embarqués sont aujourd'hui au cœur de nombreuses applications industrielles, ils sont essentiellement caractérisés par des contraintes de temps sur les actions à entreprendre qu'il faut respecter de manière plus ou moins critique. Au vu des conséquences catastrophiques (perte d'argent, de temps, ou pire de vies humaines) que pourrait entraîner une défaillance, ces systèmes doivent être sûrs de fonctionnement. La tolérance aux fautes est l'une des méthodes les plus utilisées pour assurer la sûreté de fonctionnement. Dans ce mémoire, nous avons présenté deux nouvelles heuristiques pour résoudre le problème de la génération automatique d'une distribution et d'ordonnancement temps réel qui en plus tolérante aux pannes ; Ce problème étant NP-difficile. Les deux heuristiques proposées sont basées sur la redondance des composants logiciels pour tolérer les fautes permanentes d'un seul processeur pour une architecture à liaison bus, la première se base sur la redondance active, et la deuxième sur la redondance passive. Elles offrent de bonnes performances sur des graphes d'architecture et des graphes d'algorithme générés aléatoirement.

Mots-clés : systèmes distribués temps réel embarqués, systèmes critiques, heuristiques de distribution/ordonnancement, sûreté de fonctionnement, tolérance aux fautes, fautes permanentes, redondance logicielle, architectures réparties hétérogènes.

Abstract :

The distributed embedded real time systems are today in the heart of many industrial applications, they are primarily characterized by time constraints on the actions to undertake which should be respected in a more or less critical way. Within catastrophic consequences (loss of money, time, or worse of human lives) that could involve a failure, these systems must be dependable. The fault-tolerance is one of the most methods used to ensure the dependability. In this memory, we presented two news heuristics to solve the problem of the automatic generation of a distribution and scheduling real time which in more fault-tolerant; This Problem is Np-hard. The two heuristics proposed are based on the redundancy of the software components to tolerate the permanent faults of only one processor for an architecture with multipoint communication links, the first is based on the active redundancy, and the second on the passive redundancy. They offer good performances on architecture graphs and algorithm graphs randomly generated.

Keywords: distributed embedded real time systems, critical systems, distribution and scheduling heuristic, dependability, fault-tolerance, permanent faults, software redundancy, heterogeneous distributed architectures.