Molecular Polymorphism and Virulence in *Pyrenophora tritici-repentis*

By

THEINGI SOE TAING AUNG

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of

MASTER OF SCIENCE

Department of Plant Science
University of Manitoba
Winnipeg, Manitoba

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

Molecular Polymorphism and Virulence in

Pyrenophora tritici-repentis

BY

Theingi Soe Taing Aung

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree of Master of Science

Acknowledgment

First and foremost I would like to express my gratitude to my advisor, Dr. Lakhdar Lamari for his continued support and guidance.

I would like to thank my advisory committee, Dr. G. Murray Ballance and Dr. Jim Menzies for their valuable input and suggestions.

My appreciation also goes to Dr. Brian Fristensky for his advice on phylogenetic analyses.

I am very thankful to Mr. Richard Smith and Mr. Ralph Kowatsch for their technical assistance and also their friendships.

I also appreciate the help from Mr. Bert Luit, Ms. Malgorata Balcerzak and staff from the Department of Plant Science general office. Special thanks to Ms. Martha Blouw for making me feel so welcome when I first ventured out here from Science and for her continued help throughout my masters.

Thanks to all my friends, in particular Steve Strelkov, Eymond Toupin, Brian Corbett, Ardelle Grieger, Kirsten Slusarenko, Larisa Morier and Anh Phan for all the help, good laughs and warmest friendship.

I would like to thank all those people who aided me in any way in my thesis work.

I also would like to thank NSERC for funding of this project.

Finally, I would like to acknowledge my parents who have supported me and taught me the importance of education.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figures **Page**

ABSTRACT

Tan spot is a major leaf-spotting disease of wheat, found in all the major wheat growing areas of the world. Isolates of *Pyrenophora tritici-repentis*, the causal fungus of this disease, are currently grouped into six races based on their ability to induce necrosis and chlorosis on a set of differential wheat cultivars/lines. The genetic variability of the pathogen and relationships among the six currently known races of *P. tritici-repentis* were investigated using RAPD analysis. A collection of fifty-three isolates from hexaploid and tetraploid wheats originating from Manitoba, Saskatchewan, Alberta, North Dakota, Uruguay and Algeria was used in this study. Based on 59 RAPD polymorphic bands, two distinct RAPD groups (Gp I and Gp II) with 26% similarity were obtained using the UPGMA method of analysis. Gp I included all 51 isolates of virulent races 1, 2, 3, 5 and 6 and Gp II contained the only 2 isolates of avirulent race 4 available in this study. Gp I was further divided into two subgroups, Gp IA and Gp IB, with 48% similarity. Gp IA contained all isolates of the necrosis-inducing (nec^+) races 1 and 2 and Gp IB contained all the 17 isolates of nec^- races 3, 5, 6, which are also known to induce chlorosis. Neighbor-joining distance and parsimony methods confirmed the separation between nec^+ and nec^- races. Analysis of molecular variance (AMOVA) revealed that 36% ($P<0.001$) of the total genetic diversity was attributed to the differences between nec^+ and nec^- races. Within nec^+ races, pairwise comparisons revealed that isolates of races 1 and 2 were part of the same population. The absence of genetic structure among isolates of races 1 and 2 suggested a recent origin for these two races. In contrast to the lack of genetic structure between races 1 and 2, significant

differences were detected between races 3 and 5. The results of this study strongly suggested that virulence is the major driving force in the variation and evolution of *P. tritici-repentis*.

1.0 INTRODUCTION

Pyrenophora tritici-repentis (Died.) Drechs. (anamorph *Drechslera tritici-repentis* (Died.) Shoem.), the causal agent of tan-spot of wheat is a destructive fungal pathogen which occurs throughout the major wheat growing regions worldwide (Hosford, 1982). Yield losses as high as 49% were reported for susceptible wheats when conditions favored disease development (Rees et al., 1982). *Pyrenophora tritici-repentis* has been reported to infect many gramineous species (Morrall and Howard, 1975) and is believed to have the widest host range of any *Pyrenophora* species (Shoemaker, 1962). The wide host range of *P. tritici-repentis* likely has an impact on genetic variation and population variability.

Isolates of *P. tritici-repentis* were initially categorized using a pathotype nomenclature system based on symptoms type. This system, however, is limited to a maximum of four categories: pathotype 1 (nec^+ chl^+), pathotype 2 (nec^+ chl^-), pathotype 3 (nec^-, chl^+), or pathotype 4 (nec^-, chl^-) (Lamari and Bernier, 1989b). Lamari et al., (1995) introduced race classification system to group isolates on the basis of their virulence on individual host differential genotypes. To date, six races have been identified, of which races 1 to 4 represent isolates from previous pathotypes 1 to 4 respectively. Race 5 isolates collected from eastern Algeria, are similar to race 3, except that they induce chlorosis in cv. Katepwa whereas isolates of race 3 induce chlorosis in line 6B365 but not in Katepwa. Race 6 isolates which induce chlorosis in both Katepwa and 6B365 were recently obtained from a North Africa collection (Lamari, unpublished data).

The available virulence characteristics of the different races suggested some interesting relationships among the races. Races 1 and 2 share the ability to induce necrosis

on cultivar Glenlea; races 1, 3 and 6 share the ability to induce chlorosis in line 6B365; race 5 and 6 both cause chlorosis on cultivar Katepwa. However, virulence markers may represent only a small portion of the total genetic variation present in the fungal population. No study has been carried out on the genetic relatedness of *P. tritici-repentis* races with neutral molecular markers. Random amplified polymorphic DNA (RAPD) was selected to study the genetic variation in *P. tritici-repentis*. This method has several advantages over other polymorphic DNA detection techniques such as restriction fragment length polymorpism (RFLP) and amplified fragment length polymorphism (AFLP). RAPD analysis is quick to perform, requires small amount of template DNA and does not require a prior knowledge of DNA sequence information. Notwithstanding drawbacks of this technique, RAPD markers were found to give similar results in genetic analysis of populations as RFLP, AFLP, microsatellite markers (Powell et al., 1996), and isozyme markers (Isabel et al., 1995; Nadler et al., 1995). Several researchers have successfully employed RAPD markers to study genetic and phylogenetic relationships in a wide range of plant pathogenic fungi (Chen et al., 1993; González et al., 1998; Vakalounakis and Fragkiadakis, 1999). RAPD markers can be used to characterize the genetic diversity and population structure of phytopathogenic fungi when used in statistical methods such as analysis of molecular variance (AMOVA) (Borchardt et al. 1998, Hamelin et al. 1998, Wang et al. 1997).

The objective of this study was to assess the genetic variability in *P. tritici-repentis* and to provide an insight into the genetic relationships among the six known races of this economically important wheat pathogen using RAPD procedure.

2.0 LITERATURE REVIEW

2.1 *Pyrenophora tritici-repentis*

Pyrenophora tritici-repentis, the causal agent of tan spot of wheat is a homothallic ascomycete. *P. tritici-repentis* is classified in the kingdom Fungi, division Eumycota, subdivision Ascomycotina, class Loculoascomycete, order Pleosporales, and family Pleosporaceae (Luttrel, 1973). The conidia of its anamorph, *Drechslera tritici-repentis*, are cylindrical, divided into five to seven multinucleated cells (Shoemaker, 1962; Wehmeyer,1954) and measure 95-165 x 14-18 μm. Pseudothecia of *P. tritici-repentis* are black and produce abundant eight-spored asci which are bitunicated, cylindrical and narrow at the base (Shoemaker, 1962). Ascospores are hyaline at maturity, have three transverse septa and one longitudinal septum and measure 47-65 x 20-26 μm. Conidia and ascospores spores are disseminated by wind. The pseudothecia develop on infested wheat stubble soon after harvest and mature over the fall and winter (Hosford 1971; Rees and Platz 1980; Wright and Sutton 1990). Ascospores are discharged the following spring and are generally thought to act as primary inoculum. The pathogen then spreads through repeating cycles of conidium production on diseased leaf tissue (Hosford 1972; Rees and Platz 1980).

Pyrenophora tritici-repentis has a broad host range among grass species from different parts of the world (De Wolf et al, 1998). *Pyrenophora tritici-repentis* is pathogenic on more than 33 grass species (Hosford, 1982; Krupinsky, 1992) including *Agropyron repens, A. smithii* and *Bromus inermis*. The pathogen also attacks cereal species and, based on economic importance, *Triticum aestivum* may be considered as one of the main hosts. Although *P. tritici-repentis* has been isolated from numerous gramineous species that

may function as alternative hosts for the pathogen, Howard and Morrall (1975) found low overall intensity of this disease on native prairie grasses, despite apparently favorable environmental conditions. Reproduction of *P. tritici-repentis* on these hosts may play an important role as a source of primary inoculum between growing seasons and as a source of genetic variation in the pathogen population.

Tan spot of wheat has been identified throughout the major wheat-growing regions in Australia, Asia, Africa, South America, and North America (Hosford, 1982; Wiese, 1987). Rees et al. (1982) reported yield losses as high as 49% for susceptible spring wheat when conditions favored disease development, however, losses are more commonly between 5-10% (Hosford, 1982). The disease has become a significant leaf spotting disease on the Canadian Prairies (Tekauz, 1976) and was reported as the most common disease of wheat in Manitoba in 1990 (McCallum et al., 1992). *Pyrenophora tritici-repentis* was observed and isolated from wheat in the 1930s but the fungus did not become a serious problem on wheat until the early 1970s (Hosford, 1982). The increased importance of tan spot of wheat is mainly attributed to the adoption of zero and reduced tillage which increases the amount of primary inoculum of the pathogen and allows the pathogen to survive from one season to the next. Rees and Platz (1992) reported that much of the high susceptibility to disease was observed in semi-dwarf wheats released after 1960 and changes in cultivar genotypes played an important role in the increasing importance of this disease.

The disease symptoms of tan spot consist of tan necrosis and/or extensive chlorosis on leaves (Lamari and Bernier 1989a, 1989b). Tan necrosis and chlorosis are expressed as a result of specific interactions between isolates of the pathogen and appropriate wheat

genotypes (Lamari and Bernier, 1989b). Qualitative analysis of virulence was developed by Lamari and Bernier (1989b) based on the differential induction of tan necrosis and extensive chlorosis on differential host cultivars. Lamari and Bernier, (1989b) grouped isolates of *P. tritici-repentis* into four pathotypes, based on their ability to cause tan necrosis and chlorosis (pathotype 1), necrosis only (pathotype 2), chlorosis only (pathotype 3) and neither symptoms (pathotype 4) on differential hexaploid wheat cultivars.

The identification of isolates from eastern Algeria (pathotype 3) led to the development of a race classification system to describe the new virulence pattern, on the basis of differentiation by individual cultivars. The isolates from the Algerian collection differ from those originally ascribed to pathotype 3 by their virulence (chlorosis) to cv. Katepwa and avirulence to line 6B365; the original isolates of pathotype 3 are avirulent on cv. Katepwa and virulent on line 6B365 (Lamari et al., 1995). Races 1 to 4 are currently represented by the isolates within pathotypes 1 to 4, respectively, and race 5 comprised of the Algerian isolates which induce chlorosis on cv. Katepwa. Isolates from race 5 can also be classified as pathotype 3, since they induce only chlorosis in hexaploid wheat. The recently identified race 6 isolates collected from eastern Algeria cause chlorosis on both cv. Katepwa and 6B365 line and thus, belong to pathotype 3 (L. Lamari, unpublished data). The virulence data suggest interesting relationships between some races of *P. tritici-repentis*. Races 1 and 2 share the ability to induce necrosis on cultivar Glenlea; races 1, 3 and 6 share the ability to induce chlorosis in line 6B365; race 5 and 6 both cause chlorosis on cultivar Katepwa. There are at least two factors for inducing chlorosis in race 1 isolates, one of

which is shared with race 3 and 6 and the other which is responsible for the induction of chlorosis in the tetraploid line 4B160 (Gamba and Lamari, 1998).

To date, two host-specific toxins have been isolated, purified and characterized from isolates of *P. tritici-repentis*. The Ptr necrosis toxin produced by necrosis inducing isolates in culture (Tomás and Bockus 1987) and in-planta (Lamari et al., 1995) has been shown to be responsible for the development of the necrosis symptom in susceptible wheat genotypes (Ballance et al., 1989, Lamari and Bernier, 1989c). This Ptr necrosis toxin, later referred to as Ptr ToxA, was isolated and purified from culture filtrates of necrosis inducing *P. tritici-repentis* isolates by four independent research groups (Ballance et al., 1989, Tomás et al., 1990, Tuori et al., 1995, Zhang et al., 1997). The toxin encoding gene was cloned and sequenced independently by Ballance et al. (1996) and Ciuffetti et al. (1997). From the sequence information, it was proposed that the coding region for the toxin gene encoded a 19.7 kDa proprotein and it is processed to the 13.2 kDa mature toxin protein. The comparison of nucleotide sequences from Ptr-ToxA encoding clones from different isolates from race 1 and race 2 collected in different geographical locations gave identical DNA sequences except for one base, which did not change the amino acid sequence (Ballance et al., 1996, Ciuffetti et al., 1997, G. M. Ballance, unpublished data). Southern analysis also indicated that hybridization with the toxin gene only occurred in races which induce necrosis (Ballance et al., 1996).

A second host-specific toxin, Ptr chlorosis toxin produced by race 5 isolates of *P. tritici-repentis*, was shown to be associated with development of the chlorotic symptoms in tan spot of wheat (Orolaza et al., 1995). The toxin, referred to as Ptr ToxB, was purified by

Strelkov et al. (1999) and reported as a small protein with molecular mass of 6.61 kDa. The Ptr ToxB encoding gene has been cloned from race 5 isolates and homology sequences to the Ptr-ToxB gene have been found in races 3, 4, 5 and 6 isolates based on southern analysis (G. M. Ballance, unpublished data). No homologous sequences to the Ptr-ToxB gene have been found in necrosis inducing races 1 and 2 (G.M. Ballance, umpublished data). The presence of genomic sequences related to the cloned Ptr-ToxB encoding gene in all races except races 1 and 2 suggests interesting relationships among races 3, 4, 5 and 6 of *P. tritici-repentis*. However, there has been no study published on molecular relationships among virulent races of *P. tritici-repentis* isolates, the primary focus of the present investigation.

2.2 Molecular Methods in Characterization of Fungal Pathogens

Molecular biology techniques have been increasingly used in the identification of fungi at the level of species, sub-species, variety, *formae speciale* and race (Coddington and Gould, 1992). The availability of molecular techniques has created renewed interest in the area of fungal systematics, providing effective characters for genetic, population and epidemiological studies among others (Kohn, 1992). Knowledge of genetic relationships among races and population structure of the pathogen are useful for development of new resistant cultivars and control strategies for disease epidemics. Virulence markers based on differential sets that describe genetic variation and physiological specialization in plant pathogen populations are obviously important because they provide direct information concerning the effects of selection pressures, including the host, and the potential effectiveness of resistance genes. However, virulence markers may represent only a small

portion of the total genetic variation present in the population. Molecular markers, both protein and DNA based, are generally assumed to be independent of host selection and provide new impetus for studies of the biodiversity of fungi by providing direct access to the genome.

The available molecular techniques for study of plant pathogenic fungi include protein- and DNA-based methodologies. Studies on proteinaceous components encompass immunological assays, amino acid sequencing and allozyme and isozyme analysis. Studies at the DNA level, also include a wide array of techniques such as restriction fragment length polymorphism (RFLP), amplified fragment length polymorphism (AFLP), random amplified polymorphic DNA (RAPD) and simple sequence repeats (SSR) are now available. There are advantages and disadvantages in all these molecular techniques. DNA-based methods compared with protein-based techniques have been preferred recently in characterization of different phytopathogenic fungi. It is caused by the available number of DNA markers and theoretically, DNA markers covering the entire genome can be found.

2.2.1 Protein-based Methods

Analysis of proteins have been mostly restricted to allozymes and isozymes in fungal systematics. Although detection systems for more than 57 specific enzymes in diverse eukaryotes have been established (Vallejos, 1983), less than 15 enzyme systems have been found to be useful with any particular species and in fungal species even fewer enzyme systems have been employed (Newton, 1987). There has been a number of studies on the isozyme variation within and among races and populations of plant pathogenic fungi.

Burdon and Roelfs in 1985 studied isozyme variation in asexually reproducing populations of *Puccinia graminis* and *P. recondita* of wheat. The researchers found that in *P. recondita*, there was a very low level of isozymic diversity which contrasted sharply with the high diversity of virulence races. In *P. graminis*, it was found that groups with closely related virulences resulted in the same isozyme phenotypes.

Isozyme analysis has been a useful tool in studying the genetic variation in a foliar pathogen of corn in North Carolina. Welz et al. (1993) found a set of isozyme markers for investigating genetic variation within and among pathogenic races of *Cochlibolus carbonum* on corn and used these markers to study the evolution of *C. carbonum* races.

Although isozymes markers have been proven to be a useful tool in the study of pathogenic fungi, these enzyme systems have their limitations. Limitations are that for each enzyme system optimal conditions for extraction and buffering need to be determined, intraspecific variation may be too low for detection and some enzymes are expressed at a particular developmental stage or may be tissue specific (Michelmore and Hulbert, 1987).

2.2.2 Restriction Fragment Length Polymorphism (RFLP)

The RFLP technique involves digestion of a nuclear genome with a restriction enzyme which produces many fragments. The individual bands are then identified by hybridization with a labeled homologous probe in a southern blot of the gel (Apuya et al., 1988). The RFLP approach, wherein the individual fragments can be analyzed for length variation, and in mapping approaches, both have been employed in study of fungal plant pathogenic fungi. The RFLP approach also offers the possibility of choosing sequences such

as nuclear DNA or mitochondrial DNA for analysis. However, mitochondrial DNA has been suggested to be poorly suited for RFLP analysis in fungi because of the occurrence of a high rate of deletion/insertions and rearrangements, which precludes alignment of homologous sequences for comparison of restriction enzyme sites (Dowling et al., 1990). On the other hand, Bruns et al (1991) suggested that although deletion/insertion can be potentially problematic for analysis, one major advantage is that length differences can be detected with almost any restriction enzyme, unlike site changes which are exclusive to a particular enzyme.

Nuclear RFLPs in pathogenic fungi have been used to develop linkage maps and to study pathogen population genetics. McDonald and Martinez (1990) used RFLP to study the distribution of genetic variation in a population of *Mycosphaerella graminicola*. The same RFLP technique was employed in studying gene flow between geographic location and genetic structure of populations of *M. graminicola* (Boeger et al 1993; Chen and McDonald, 1996). These studies have found evidence that gene flow occurring over very long distances, and that sexual reproduction plays a major role in the genetic structure of populations, of which could have significant implications for control strategies and plant breeding programs. Genetic variability and clonal distribution of *Sclerotinia sclerotiorum* was also studied using the RFLP method and results suggest that field populations of *S. sclerotiorum* are genetically heterogeneous and clones of *S. sclerotiorum* are distributed over long distances geographically (Kohn et al, 1991; Kohli et al, 1992).

The application of RFLP in studying genetic variation and the population structure of phytopathogenic fungi have provided us with useful information on disease cycles and

epidemiology. The use of RFLP is, however, not without its limitations. First of all, large amount of good quality DNA are required for restriction enzyme digestion. Secondly, the restriction enzymes do not perform well if the DNA is contaminated with proteins or other chemicals such as phenol or chloroform that may have been used during the DNA extraction process. Thirdly, if the analysis includes hybridization with radiolabelled probes, the procedure becomes laborious and lengthy, in addition to safety concerns involving the use of radioisotopes.

2.2.4 Amplified Fragment Length Polymorphic DNA (AFLP)

The AFLP technique is based on the selective PCR amplification of restriction fragments from a total digest of genomic DNA. The technique involves three steps: (i) restriction of DNA and ligation of oligonucleotide adapters, (ii) selective amplification of sets of restriction fragments, and (iii) gel analysis of the amplified fragments (Vos et al., 1995). The AFLP technique is used to visualize DNA polymorphism between samples, determine the identity of a specific DNA sample, or to assess the relatedness between samples. It can also be used as the source for genetic markers to generate linkage maps or to identify molecular markers linked to phenotypic traits and/or genetic loci. AFLP has been applied in studying the genetic variation of plant pathogenic fungi in recent years. Fungal pathogens such as *Phytophthora infestans, Colletotrichum* species of alfalfa and *Colletotrichum lindemuthianum* were studied using AFLP techniques (Van der Lee et al., 1997, O'Neill et al., 1997, Gonzalex et al., 1998).

The AFLP analysis of *C. lindemuthianum* from Mexico determined the genetic

distances and produce a dendrogram which demonstrated two levels of association based on the type of cultivar from which isolates of *C. lingemuthinum* originate and also differentiated subgroups based on the geographic location from which they were obtained (Gonzalex et al., 1998). Van der Lee et al., (1997) constructed a linkage map based on AFLP markers in *P. infestans*. *C. trifolii* and *C. gloeosporiodes* were identified from other *Colletotrichum* species of alfalfa based on AFLP markers by O'Neill et al (1997). The AFLP technique proved to be a useful tool in the study of the genetic structure and variation in fungal pathogens, since, the high sensitivity of AFLPs provides more information on variation than any other molecular technique (Majer et al., 1996). The drawbacks of this technique, however, is it requires good quality DNA for restriction enzyme digestion and level of technical skill required is high.

2.2.5 Simple Sequence Repeats (SSR) or microsatellites

Throughout the genome of most organisms, apparently at random, are regions of short repeats of di- or trinucleotides termed SSRs or microsatellites (Provan et al., 1996). Replication of such regions is often subject to 'slippage' by the DNA polymerase, resulting in different lengths of repeats which may be regarded as allelic (Duncan et al., 1998). Primers which span such polymorphic regions can be designed for PCR amplification. The amplification products from SSR primers can be visualized following electrophoresis in polyacrylamide gels with radioactive labelling of primers or agarose gels with ethidium bromide staining. This technique allows allele segregation to be analyzed, and is widely used for studies of gene flow and linkage mapping (White and Powell, 1997). There have been

some studies on molecular diversity of fungi by amplification of microsatellite regions. Longato and Bonfante (1997) demonstrated the presence of microsatellite sequences in the genomes of ecto- and endo- mycorrhizal fungi and discriminated between mycorrhizal symbionts with different taxonomic features. In principle, SSR could be used in screening of virulence races in pathogenic fungi and has a potential of being a widely used technique for studies of fungal population genetics. In order to use microsatellites markers for population studies, a prior knowledge of the genome under study is needed to design SSRs primers. RAPDs and AFLPs do not require a prior knowledge of the genome, therefore, requires less development than SSRs and yield many more polymorphic bands. However, microsatellites technique produce more reproducible bands than RAPD, more cost-effective than RFLP and simple to use. It represents an alternative to RFLP markers for tagging genes controlling agronomic characters and is a simple and reliable tool for assaying genetic diversity in host and pathogen populations.

2.2.6 Molecular Markers with Organelle DNA

Mitochondrial DNA encodes some of the proteins essential for the functioning of mitochondria, but large segments of its genome appear to have little or no coding function. In many organisms, mtDNA is inherited maternally without contribution from the paternal line and with no genetic recombination, making it a valuable complement to genomic DNA, in which recombination occurs as a result of sexual reproduction. It is very likely that mtDNA is inherited maternally in many, if not all fungi. The mtDNA mutates much more rapidly than does genomic DNA but at a relatively constant rate across many organisms,

making it ideal for epidemiological studies (Day and Shattock, 1997) and for looking at closely related species or isolates within a species. Probes for RFLPs, derived from purified mtDNA, have been used to characterize mtDNA types. RFLPs with mtDNA have been used to study relationships between *Phytophthora cryptogea* and *P. drechsleri* (Mills et al., 1991), and within *P. megasperma* (Förster and Coffey, 1993), *P. porri* (De Cock et al., 1992) and *P. fragariae* (Stammler et al. 1993). Typing of mtDNA of *P. infestans* (Carter et al., 1990) has been used for epidemiological studies (Day and Shattock, 1997).

Ribosomal RNA (rDNA) gene repeat encode the rRNA species which are incorporated into ribosomes, the sites of protein synthesis in the cell. The rDNA array of an eukaryotic nuclear genome typically consists of several hundred tandemly repeated copies of the transcription unit and nontranscribed spacer: the intergenic spacer (IGS), the internally transcribed (ITS) spacers ITS1 and ITS2 and the externally transcribed spacer (ETS). Unlike subunit sequences, spacer regions have no apparent function and they are less conserved. Mutations are assumed to be neutral and to occur at a fairly constant rate, and differences accumulated over time have been used to study relatedness among taxa below the level of Orders: genera, species and sub-species in particular (Cooke and Duncan, 1997). ITS1 and ITS2 sequences have been used as the basis of highly sensitive PCR-based detection or diagnostic tests for a number of fungal pathogens including *Phytophthora* (Bonants et al. 1997). Although useful for distinguishing among species, ITS regions have limited values as markers for sexual recombination within a species since there is little or no variation at that level. Combined with RAPD analysis, rDNA markers have been employed in study of genetic relationships among the four species of wheat bunt fungi (Shi et al., 1996).

2.2.7 Random Amplified Polymorphic DNA (RAPD)

Genetic fingerprinting using random amplified polymorphic DNA (RAPD) is a powerful molecular tool used to differentiate morphologically similar microorganisms (Welsh and McClelland, 1990; Williams et al., 1990). The RAPD procedure detects nucleotide sequence polymorphisms in a DNA amplification-based assay using only a single primer of arbitrary nucleotide sequence. In this reaction, a single species of 10 base (decamer) primer anneals to the genomic DNA at two different sites on the opposite strands of the DNA template. If these priming sites are within an amplifiable distance of each other, a discrete DNA product is produced through PCR amplification. The presence of each amplification product identifies complete or partial nucleotide sequence homology, between the genomic DNA and the oligonucleotide primer, at each end of the amplified product. On average, each primer will direct the amplification of several discrete loci in the genome, making the assay an efficient way to screen for nucleotide sequence polymorphism between individuals which means that there is a very good chance of finding RAPD polymorphism in different species. For example, the frequency of finding RAPD polymorphisms has been shown to be 0.3 per primer in *Arabidopsis thaliana* and 2.5 per primer in *Neurospora crassa* (Tingey and del Tufo, 1993). The RAPD method has several technical advantages such as the reduced amounts of DNA needed and the ability to simultaneously screen many samples. Numerous experiments are possible in a short time and data are relatively easy to obtain if the polymorphisms are few. RAPD markers have been used in development of genetic maps, identify markers linked to a trait of interest, pooling strategies and population genetics.

The area of research that has shown the most growth with respect to the use of RAPD technology is that of population genetics (Hedrick, 1992). Several groups have reported on the utility of RAPD markers as a source of phylogenetic information. RAPD markers have been used effectively to assess the amount of genetic diversity and the relationship among virulent races in phytopathogenic fungi such as *Puccinia striiformis* f. sp. *tritici* (Chen et al., 1993), *Puccinia recondita* f. sp. *tritici* (Kolmer et al., 1995), *Setosphaeria turcica* (Borchardt et al., 1998), *Colletotrichum lindemuthianum* (González et al., 1998) and *Cronartium ribicola* (Hamelin et al. 1998). RAPD markers were also used to identify and discriminate pathotypes of *Fusarium oxysporum* sp. *ciceris* (Kelly et al., 1994) and *Leptosphaeria maculans* (Goodwin and Annis, 1991); races of *Fusarium oxysporum* f. sp. *pisi* (Grajal-Martin et al., 1993) and *Fusarium oxysporum* f. sp. *lycopersici* (Mes et al., 1999). Vakalounakis and Fragkiadakis (1999) distinguished the isolates of *F. oxysporum* f. sp. *radicis-cucumerinum* from those of *F. oxysporum* f. sp. *cucumerinum* with RAPD markers. Weiland et al., (1999) identified five RAPD markers associated with low virulence in *Pyrenophora teres* f. *teres*. Although, RAPD markers have been successfully employed in assessing genetic variation in plant pathogen populations, there are a few drawbacks with the technique, which include reproducibility and fewer detectable polymorphisms than AFLP. Not withstanding these criticisms and limitiations, RAPD markers have been found to give the same results in genetic analysis of populations as RFLP, AFLP and SSRs (Powell et al., 1996), as well as isozyme markers (Isabel et al., 1995; Nadler et al., 1995).

2.3 Phylogenetic Analysis

Phylogenetics is the area of research concerned with finding the genetic connections and relationships between species. Phylogenetic analysis of molecular data can be used to identify individuals or populations which may have arisen from a common origin, and those which are mostly closely related. These relationships are visualized in the form of a phylogenetic tree. Several methods are available for the phylogenetic analysis of molecular data such as RFLPs, RAPD or sequence data (Holsinger and Jansen 1993; Swofford and Olsen 1990; Hillis et al., 1993). There is no ideal method for any type of data and each phylogenetic method has its own set of assumptions, strengths and weaknesses. Distance, parsimony and maximum likelihood methods are three general approaches to phylogenetic analysis of molecular data. Distance and parsimony methods have been commonly used to study the relationships among isolates and pathotypes of plant pathogenic fungi based on RAPD data.

With distance methods, the character state data are summerized in a distance coefficient matrix which relates all pairs of taxa (Holsinger and Jansen 1993). A variety of methods have been proposed for phylogenetic analysis of distance data. The most widely uses are UPGMA cluster analysis (unweighted pair group method using arithmetic averages), Fitch-Margoliash (a form of weighted least squares), and neighbor joining (Holsinger and Jansen 1993). UPGMA is the most widely used method of all three phylogenetic analysis partly because the method usually generates satisfactory results and also it is computationally efficient. UPGMA has been chosen in the majority of genetic studies with phytopathogenic fungi using RAPD data such as in *Puccinia striiformis* f. sp.

tritici (Chen et al., 1993), *Puccinia recondita* f. sp. *tritici* (Kolmer et al., 1995) and *Colletotrichum lindemuthianum* (González et al., 1998). UPGMA constructs a tree by successive clustering using an average-linkage method of clustering. The method assumes a constant rate of evolutionary change and that mutations occur at some regular, more or less predictable rate (molecular clock). Fitch-Margoliash and neighbor-joining analyses do not require the assumption of a constant rate of substitutions, but simulations have suggested that neighbor-joining is better at obtaining the correct phylogeny and also it is more computationally efficient than Fitch-Margoliash (Holsinger and Jansen 1993). The neighbor-joining method constructs a tree by successive clustering of lineages, setting branch lengths as the lineages join and the process continues until two groups or nodes are left separated by a single branch. The genetic relationships between the various morphological groups within *Colletotrichum acutatum sensu lato* was represented by neighbor-joining distance tree by Lardner et al. (1999).

Although distance methods have certain advantages for the analysis of RAPD data, notably their computational efficiency, they also have an important disadvantage. By reducing the data to pairwise distances, important information on the evolutionary history of individual polymorphic site is lost. The goal of parsimony is to select the best tree (or trees) which require the minimum amount of evolutionary changes from the set of all possible trees. There are a few parsimony methods available and three of the well known methods are Dollo (Farris et al., 1977), Wagner (Farris et al., 1970) and Polymorphism parsimony (Felsenstein et al., 1979). The methods differ from one another in how the amount of evolutionary change is calculated. Wagner parsimony presumes that the presence

of a band (character state 1) is as likely as the absence of that band (character state 0). The method counts each change of site, from presence to absence or absence to presence, as a single step, and minimizing the length of the tree under the Wagner criterion corresponds to minimizing the total number of steps on the tree. Wagner parsimony phylogenetic analysis was used to represent the genetic diversity and genetic relatedness among the *Fusarium oxysporum* f. sp. *pisi* (Grajal-Martin et al., 1993) and *Fusarium oxysporum* isolates from cucumber (Vakalounakis and Fragkiadakis, 1999) based on RAPD data. Dollo parsimony, on the other hand, assumes that presence of a band (character state 1) is so unlikely relative to its loss (character state 0) that any taxa sharing a particular site must have inherited it from a common ancestor. Thus, character state 1 can be gained only once, but it may be lost many times; minimizing the total number of band losses while allowing each band to be gained only once. Based on Dollo parsimony, if polymorphism for a given site exists in a population (i.e. both 1 and 0 are present), a population containing both alleles might be sampled and erroneously assigned 0, even though 1 is present in other individuals that were not sampled. Polymorphism parsimony tries to take this possibility into account. If most nodes on a branch contain 0s, but at least one 1 node is implied, the polymorphism parsimony option of Dollo parsimony will assume that the presence of the 1 is the result of polymorphism in the population, rather than a rare forward mutation.

The another promising approach for the phylogenetic analysis of RAPD data is to use a statistical model of presence of absence of bands to construct a maximum-likelihood estimate. Maximum-likelihood model (Felsenstein, 1992) is perhaps the most promising approach for the phylogenetic analysis of restriction site data (Holsinger and Jansen 1993)

and also another promising model for RAPD data. The procedure relies on evolution models that are quite sophisticated, taking into account the possibility of unequal rates of change among lineages, site-specific rate variability, and the random distribution and/or clustering of variable sites. There is one difficulty in applying the maximum-likelihood method to large data sets, especially those with a large number of taxa (Holsinger and Jansen 1993). It requires massive amounts of computer power and is the least computationally efficient of any of the methods discussed in this section.

2.4 Analysis of Molecular Variance (AMOVA)

Analysis of Molecular Variance (AMOVA) is a method for studying molecular variation within a species. AMOVA uses a classical analysis of variance (ANOVA) approach to compute molecular variance components at different hierarchical levels. Information on DNA haplotype divergence is incorporated into an analysis of variance format, derived from a matrix of squared-distance among all pairs of haplotypes. The AMOVA produces estimates of variance components and Wright's F-statistics analogs, designated as Φ-statistics, reflecting the correlation of haplotypic diversity at different levels of hierarchical subdivision. The significance of the variance components and Φ-statistics is tested using a permutational approach, eliminating the normal distribution assumption that is conventional for analysis of variance but inappropriate for molecular data (Excoffier, et al 1992).

The analysis of molecular data using AMOVA method had been applied in the fields of human genetics, zoology, ecology and plant pathology. AMOVA was applied to human

mtDNA restriction data by Excoffier et al. (1992) to find significant differences among human populations within and between regions of the world. RAPD markers were analyzed by AMOVA in study of genetic variation in buffalograss (Huff et al., 1993) and molecular variation within and among populations of perennial ryegrass (Huff 1997). McMillan and Bermingham (1996) used AMOVA to find a significant difference among the populations of Dall's porpoise in the Northern and Western North Pacific. The measure of genetic variation in sampled populations of plant pathogenic fungi has also been accomplished by AMOVA analysis. AMOVA was used in the study of genetic diversity and population structure of *Gremmeniella abietina*, canker pathogen fungus from *Pinus contora* (Wang et al., 1997), and *Cronartium ribicola*, white pine blister (Hamelin et al., 1998).

3.0 MATERIALS AND METHODS

3.1 Fungal Isolates.

Fifty-three single-spore isolates representing the known six races of *P. tritici-repentis* collected off wheat from Manitoba, Alberta, Saskatchewan, North Dakota, Uruguay and Algeria were selected for the study (Table 1). The isolates were selected based on geographic origin, year of collection and host from which they were collected. The number of isolates from each race was selected based on the availability of isolates in the U of Manitoba collection. Altogether, 33 isolates of race 1 and race 2 were included in this study, 27 of which originated from hexaploid wheats and six from durum wheats. The other 20 isolates consisted of isolates of races 3, 5 and 6 which were originally obtained from durum wheats. Conidia of *P. tritici-repentis* were produced from monoconidial isolates on V-8 potato-dextrose agar (V8-PDA) medium as described in Lamari and Bernier (1989a) and Appendix 1A.

3.2 Inoculum Production and Inoculation.

Cultures were incubated in the dark for 5 days at 20°C until they were approximately 4 cm in diameter. The cultures were then flooded with sterile distilled water, the mycelium flattened with the bottom of a flamed test tube, and excess water was decanted. Plates were placed under intense lighting for 18 h at 20-24°C, followed by 18-24 h at 15°C in the dark. Conidia were harvested by flooding the Petri dishes with sterile distilled water and dislodging the spores with a wire loop. Two or three additional water rinses of cultures

Table 1. Isolates of *P. tritici-repentis* used in this study.

Isolate No.	Isolate Name	Race	Year Isolated	Original Wheat Host	Place of Origin
1	ASCI	1	1985	Hexaploid	Manitoba
2	98MD5	1	1998	Hexaploid	Manitoba
3	98MD7	1	1998	Hexaploid	Manitoba
4	98MD8	1	1998	Hexaploid	Manitoba
5	98MD10	1	1998	Hexaploid	Manitoba
6	98MD11	1	1998	Hexaploid	Manitoba
7	94-12	1	1994	Hexaploid	Manitoba
8	11JA	1	1992	Hexaploid	Alberta
9	15JA	1	1992	Hexaploid	Alberta
10	4JS	1	1992	Hexaploid	Saskatchewan
11	Jeff-pt98	1	1998	Hexaploid	Manitoba
12	PDY7	1	<1982	Hexaploid	North Dakota
13	FH86	1	(1986)?	Hexaploid	Kansas
14	85-3	2	1985	Tetraploid	Manitoba
15	91-18	2	1991	Hexaploid	Manitoba
16	44F	2	1996	Hexaploid	Uruguay
17	32JA	2	1992	Hexaploid	Alberta
18	92-40J	2	1992	Hexaploid	Manitoba
19	91-52	2	1991	Hexaploid	Manitoba
20	90-27	2	1990	Hexaploid	Manitoba
21	92-28J	2	1992	Hexaploid	Manitoba
22	94-106	2	1994	Hexaploid	Manitoba
23	90-68	2	1990	Hexaploid	Manitoba
24	91-46	2	1991	Hexaploid	Manitoba
25	92-43	2	1992	Hexaploid	Manitoba

Isolate No.	Isolate Name	Race	Year Isolated	Original Wheat Host	Place of Origin
26	94-9M	2	1994	Hexaploid	Manitoba
27	86-124	2	1986	Hexaploid	Manitoba
28	331-9	3	1985	Tetraploid	Manitoba
29	331-2	3	1985	Tetraploid	Manitoba
30	94-25	3	1994	Tetraploid	Manitoba
31	94-115	3	1994	Tetraploid	Manitoba
32	D308	3	1985	Tetraploid	Manitoba
33	90-2	4	1990	Hexaploid	Manitoba
34	92-171	1	1992	Hexaploid	Saskatchewan
35	49JA	4	1993	Hexaploid	Alberta
36	Alg3-24	5	1993	Tetraploid	Algeria
37	Alg3-X3	5	1993	Tetraploid	Algeria
38	Alg3-X1	5	1993	Tetraploid	Algeria
39	Alg4-X1	5	1993	Tetraploid	Algeria
40	Alg7-X3	5	1993	Tetraploid	Algeria
41	Alg5-X1	5	1993	Tetraploid	Algeria
42	AlgH1	6	1996	Tetraploid	Algeria
43	AlgH2	6	1996	Tetraploid	Algeria
44	SC24-3	1	1999	Tetraploid	Saskatchewan
45	SC29-2	1	1999	Tetraploid	Saskatchewan
46	SC10-1	1	1999	Tetraploid	Saskatchewan
47	SC3-11	2	1999	Tetraploid	Saskatchewan
48	SC3-4	2	1999	Tetraploid	Saskatchewan
49	SC24-2	2	1999	Tetraploid	Saskatchewan
50	SC22-2	3	1999	Tetraploid	Saskatchewan

Isolate No.	Isolate Name	Race	Year Isolated	Original Wheat Host	Place of Origin
51	SC29-1	3	1999	Tetraploid	Saskatchewan
52	SC29-9	3	1999	Tetraploid	Saskatchewan
53	SC29-8	3	1999	Tetraploid	Saskatchewan

were made to resuspend and recover the conidia that had settled. Inoculum concentration was adjusted to 3500 conidia/ mL using a cell counter (Hausser Scientific, Blue Bell, Pa.). One drop of Tween 20 (polyoxyethylene sorbitan monolaurate) were added per 100 mL of conidial suspension. Seedlings of each of the differential cultivars (Table 2) were sprayed at the 2-leaf stage until runoff with the conidial suspension and incubated for 24 h under continuous leaf wetness at 20°C and a 16 h photoperiod (Lamari and Bernier 1989a). The plants were transferred to a growth room bench and observed daily for symptom development. Duplicate sets of differential lines (4 to 6 seedlings for each cultivar/line) were inoculated in each test. Virulence assessment of the single-spore isolates were determined as described previously by Lamari and Bernier (1989a). Resistance was characterized by the presence of small, brown to black spot with very little or no chlorosis or tan necrosis, and susceptibility by the presence of either symptoms. All isolates used in this study were tested for characteristic virulence pattern on a set of five differential wheat lines/cultivars (Table 2).

Table 2. The pathotype/race classification in *P. tritici-repentis* based on reaction of five haxaploid wheat lines/cultivars.

		Reaction of differentials				
Race	Pathotype	Katepwa	Glenlea	6B365	Salamouni	6B662
1	1	S (N)	S (N)	S (C)	R	R
2	2	S (N)	S (N)	R	R	R
3	3	R	R	S (C)	R	R
4	4	R	R	R	R	R
5	3	S (C)	R	R	R	S(C)
6	3	S (C)	R	S (C)	R	S (C)

R = resistant; S = susceptible; C = chlorosis; N = necrosis

3.3 Fungal Cultures for DNA Extraction.

Cultures of *P. tritici-repentis* were grown on V8-PDA (Lamari and Bernier, 1989a and appendix 1a) for approximately 5 days until they were 4 to 5 cm in diameter. Five plugs (1 cm in diameter) were cut from each culture and transferred to 250 mL Erlenmeyer flask containing 75 ml of Fries liquid medium amended with 0.1% yeast extract (Dhingra and Sinclair, 1985 and appendix 1b). Cultures were grown for 5 days in the dark on a shaker (approximately 150 RPM) at room temperature (22-25°C). Mycelial mats were harvested by filtration through Whatman paper #3. The harvested mycelium was placed in liquid nitrogen and stored overnight at -70°C, lyophilized and kept at -20°C until processed.

3.4 DNA Extraction.

Genomic DNA was extracted from mycelium using a modification of the method of White (White and Kaper, 1989). Approximately 0.5 gram of mycelium was ground under liquid nitrogen and suspended in 50:50 mix of extraction buffer (0.1 M glycine, pH 9.0, 50 mM NaCl, 10 mM EDTA, 2% SDS, 1% sodium lauryl sarcosine) (Appendix 2) and phenol/chloroform/isoamyl alcohol (25:24:1). Samples were centrifuged for 15 minutes at 6,000 rpm and the upper aqueous phase was recovered. The DNA was precipitated by addition of an equal volume of cold isopropanol. After the first precipitation with isopropanol for at least 2 hr in -20°C, the DNA was washed with 70% ethanol and the DNA pallet was air dried for approximately 1.5 hr. The DNA pellet is redissolved in 5 ml of extraction buffer and re-extracted with an equal volume of phenol/chloroform/isoamyl alchol (25:24:1). RNase adjusted to a final concentration of 150 µg/ml was added to DNA solution and incubate for 1 h at 37°C . DNA was precipitated again by adding 1/10 volume of 3M NaAc and 2 volumes of cold 95% ethanol. DNA was pelleted by centrifugation at low speed around 750 rpm. The DNA pellet was air dried until it was totally dried which usually took about 2-3 hr, and resuspended in TE buffer. The concentration of DNA was measured using a spectrophotometer at O.D. 260 nm and the final concentration adjusted to 10 ng/µl. Approximately 5 µg of DNA was electrophoresed in 1% agarose gel with 1X TAE buffer and ethidium bromide (EtBr) staining to check for DNA degradation and RNA contamination (Fig. 1). The DNA solution was stored at 4°C for RAPD analysis.

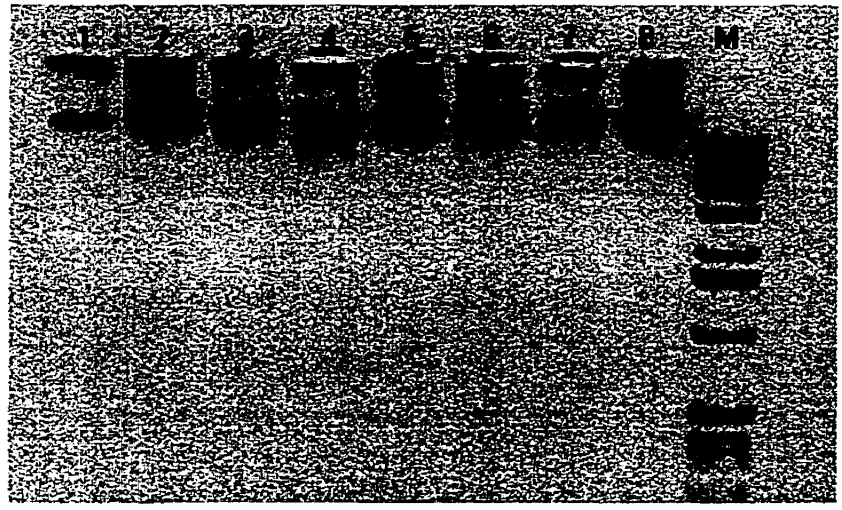


Fig. 1. Extracted genomic DNA from *P. tritici-repentis* isolates. Lane 1, isolate PDY7; Lane 2, isolate 92-43; Lane 3, isolate 94-115; Lane 4, isolate 4JS; Lane 5, isolate Alg3X1; Lane 6, isolate 94-25; Lane 7, isolate FH86; Lane 8, 85-3 and Lane M, 1Kbp DNA ladder. The extracted genomic DNA was run on 1% agarose gel with ethidium bromide (EtBr) staining to visualize DNA degradation and/or RNA contamination. Note: DNA degradation and RNA contamination were not detected in extracted DNA.

3.5 PCR Amplification Condition.

Each amplification reaction was performed in a 25 µl volume consisting of 2.5 µl of 10X Taq polymerase reaction buffer (Gibco BRL), 1.0 µl of 50 mM $MgCl_2$ (Gibco BRL), 1.0 µl of a 5 mM stock each of dATP, dCTP, dGTP, and dTTP, 0.25 µl of Taq polymerase (5 units/µl) (Gibco BRL), 0.5 µl of 15 µM RAPD primer, 2.5 µl of 10 ng/µl of template DNA and 17.25 µl of sterile HPLC-grade water. RAPD primer set #600 consisting of 54 primers used in these experiments was purchased from the department of Microbiology, University of British Columbia, Vancouver, B.C. Amplification was carried out in a thermal cycler PTC-100 (MJ researcher) programmed for 5 min at 95°C, followed by 40 cycles of 0.5 min at 95°C, 1 min at 34°C followed by 1.5 min at 72°C. PCR products were electrophoresed in 1.2% agarose gel with 1X TAE running buffer and visualized with ethidium bromide staining. Digital images of the gels were captured using a Fluor-S multilmager system PC from BIORAD, printed and archived electronically.

3.6 RAPD Analysis.

Fifty-four primers from the primer set #600 were initially screened for amplification of polymorphic bands with the *P. tritici-repentis* isolates (Table 1). Eight primers (601, CCGCCCACTG; 602, GCGAAGACTA; 609, ACAGCACCAT; 638, GCGGTGACTA; 648, GCACGCGAGA; 649, AATGCTGGAC; 650, AGTATGCAGC; 652, CCCAACACAC) were found to produce reproducible polymorphic bands. The DNA from 53 isolates of *P. tritici- repentis* was amplified using eight selected primers. Only bands

reproduced with the same primer in at least two experiments conducted at different times were used in data analysis.

3.7 Phylogenetic Analysis.

DNA bands generated by RAPD analysis were scored as "1" for the presence of a specific size amplification product and "0" for the absence of a corresponding sized band (Appendix 5) from the hard copy of the gel digital image. The scoring was also confirmed by subjecting digital images of the gels to analysis using Eagle Sight software (version 3.2; Stratagene). Bands of the same size, generated from *P. tritici-repentis* isolates by a single primer, were considered identical. The binary data sets ("0" and "1") were pooled for each isolate separately and used to calculate genetic distances based on Jacard's coefficient. A similarity matrix based on Jacard coefficient was generated by the SIMQUAL program of NTSYSpc (version 2.0) (Exeter Software, Setauket NY). Jacard coefficient (Jacard, 1908) is defined as:

$$[1]\ S_{ij} = a/(a+b+c+d)$$

where S_{ij} = Jacard similarity coefficient, a = number of 1-1 matches, b = number of 1-0 matches, c = number of 0-1 matches, and d = number of 0-0 matches.

A dendrogram was derived from the similarity matrix with the SHAN program in NTSYSpc (version 2.0). The unweighted pair group arithmetic mean method (UPGMA) was used. A neighbor joining distance radial tree was generated by Neighbor program in PHYLIP (Phylogeny Inference Package) Version 3.5c by J. Felsenstein (1993). To produce the polymorphism parsimony tree for the entire data set, one hundred bootstrapped data sets

were generated from the initial binary data set using SEQBOOT. Parsimony trees were then generated using the DOLLOP (Parsimony method), and a consensus tree was derived using CONSENSE.

3.8 Analysis of Molecular Variance (AMOVA).

AMOVA analysis was performed on variance between necrosis-inducing races (nec^+) and non-necrosis (nec^-) but chlorosis inducing races. AMOVA analysis between virulent races (races 1, 2, 3, 5, 6) and avirulent race (race 4) was not calculated because of the small sample size (2 isolates) of race 4. The genetic distance, calculated as the Euclidean distance (Excoffier et al. 1992), was computed for all pairs of the 51 isolates. The Euclidean's metric distance between two isolates is defined by Excoffier et al. (1992) as

$$[2]\quad E = n\,(1 - n_{xy}/n\,)$$

where n is the total number of polymorphic fragments and n_{xy} the number of fragments shared by the isolates x and y. The Euclidean metric distance between any two isolates is equivalent to the number of observed band differences out of the total number of markers surveyed. The analysis of molecular variance (AMOVA) (version 1.55) procedure (Excoffier et al., 1992) was used to estimate variance components for RAPD genotypes and for partitioning the variation among isolates within groups, races and among races. The analysis was performed on a matrix of squared molecular distances between pairs of observations, in our case RAPD haplotypes. A conventional sum of squares was calculated as the sum of squared deviations (SSD) between individual observations:

$$[3]\quad SSD_{(total)} = SSD_{(among\ groups)} + SSD_{(among\ races\ within\ groups)} + SSD_{(within\ races)}$$

and the corresponding mean squared deviations (MSD) were obtained by dividing the SSD by the appropriate degrees of freedom. Variance components were obtained by equating the MSD with its expectations as σ^2_a (among groups), σ^2_b (among races within groups), and σ^2_c (within races). The variance components from the analysis are used to estimate Φ statistics which are similar to F statistics and used to test for significance.

Φ values were calculated as:

[4] $\Phi_{ST} = \sigma^2_a + \sigma^2_b/(\sigma^2_a + \sigma^2_b + \sigma^2_c)$, $\Phi_{CT} = \sigma^2_a/(\sigma^2_a + \sigma^2_b + \sigma^2_c)$, and $\Phi_{SC} = \sigma^2_b/(\sigma^2_a + \sigma^2_b)$

where Φ_{ST} is the total correlation between random haplotypes within races and random haplotypes drawn from all races, Φ_{CT} is the correlation between random haplotypes within groups and random haplotypes from all races, and Φ_{SC} is the correlation between random haplotypes within races and random haplotypes within the corresponding group. Nonparametric significance tests of Φ_{ST}, Φ_{CT} and Φ_{SC} were conducted with 1000 random permutations of the data set. Under the null hypothesis, samples are considered as drawn from a global population, with variation due to random sampling in the construction of populations. Null distributions were generated by allocating every individual to a randomly chosen population while holding the sample sizes constant over a large number of permutations (N = 1000). Probabilities of observing random genetic distances and variance components greater than those generated in the analysis were computed. AMOVA evaluated the five races of *P. tritici-repentis* in two ways (Appendix 3, 4). A nested analysis which was described above and analysis among two groups where races were lumped into two groups (nec^+ and nec^-). For analysis among nec^+ and nec^- groups, a sum of squares (SSD) was calculated as:

$$[5]\ SSD_{(total)} = SSD_{(among\ groups)} + SSD_{(among\ races\ within\ groups)}$$

and the corresponding mean squared deviations (MSD) were obtained by dividing the SSD by the appropriate degree of freedom. Variance components were obtained by equating the MSD with its expectations as σ^2_a (among groups) and σ^2_b (among races within groups). Only Φ_{ST} was calculated by AMOVA since this test was only for variation among two groups. Φ_{ST} was defined as:

$$[6]\ \Phi_{ST} = \sigma^2_a / (\sigma^2_a + \sigma^2_b)$$

Pairwise population comparisons examined with AMOVA produce values of Φ_{ST} that are interpreted as the average interpopulation distance between any two populations. To determine which pairs of populations were different or similar with respect to RAPD genotypes, genetic distances (analogous to a coefficient of coancestry) were also calculated by AMOVA. Levels of significance for variance components, genetic distances and coancestry coefficients were computed by a non parametric permutational procedure using the WINAMOVA program (Excoffier et al., 1992) with 1000 permutations.

4.0 RESULTS AND DISCUSSION

4.1 Screening of Primers and Analysis of RAPD Pattern

The polymerase chain reaction (PCR) was first performed with 54 primers on 10 isolates (for primer 601) and 11 isolates (for other primers) from six races of *P. tritici-repentis*. Primers which produced polymorphic bands were selected for RAPD analysis (Fig. 2, 3, 4, 5). The primers which failed to produce any polymorphic bands among isolates (eg. Primer 651 in Fig. 3) or produce any amplification products (eg. Primer 653 in Fig. 4) were not selected. Initially, ten primers (601, 602, 609, 612, 638, 648, 649, 650, 652, 654) were selected for RAPD analysis with 53 isolates. Although primer 612 (5'-CCGTGAGTAT-3') and primer 654 (5'-CCCTGGTCTG-3') produced polymorphic bands when screened with 11 *P. tritici-repentis* isolates (Fig. 4, 5), they failed to produce reproducible bands when tested with all 53 isolates. Therefore, eight primers were selected and used to screen the DNA of the 53 isolates for polymorphic and reproducible bands. A total of 59 scorable polymorphic RAPD bands were found among the 53 isolates from 6 races (Table 3). Some primers, such as primer 650, produced a moderate number of polymorphic bands and were relatively easy to score (Fig. 6). However, amplification of total DNA with primers such as 652 resulted in complex fingerprint patterns and the size differences in base pairs among bands were very close, making it hard to score all the polymorphic bands (Fig. 7). The high level of polymorphism produced by primer 652 suggested that there might be a high number of sequence repeats homologous to primer 652 sequence (CCCAACACAC) in the pathogen genome.

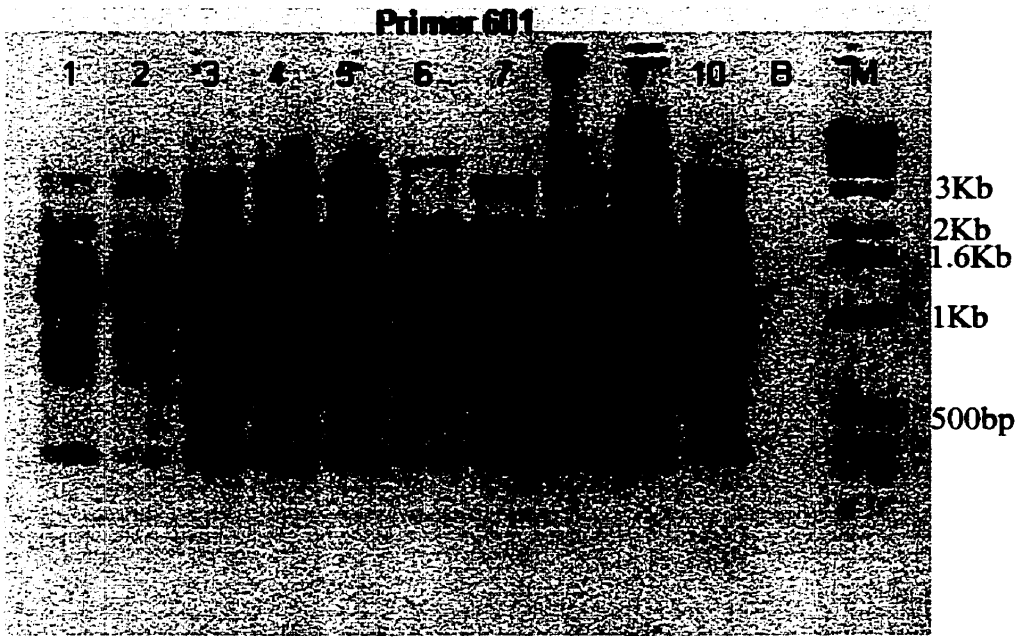


Fig 2. DNA amplification bands obtained with primers 601 on 10 isolates of *P. tritici-repentis*. Lane marked M are 1-Kbp ladder DNA and those marked B are control with sterile water replacing template DNA.
Lane 1, isolate ASCI (race 1); The lane 2, isolate 11JA (race 1); Lane 3, isolate 94-83 (race 2); Lane 4, isolate 32JA (race 2); Lane 5, isolate 94-115 (race 3); Lane 6, isolate 331-2 (race 3); Lane 7, isolate 90-2 (race 4); Lane 8, isolate 49JA (race 4); Lane 9, isolate Alg3-24 (race 5); Lane 11, AlgH1 (race 6).
Note: Primer 601 produced polymorphic bands and selected for RAPD analysis.

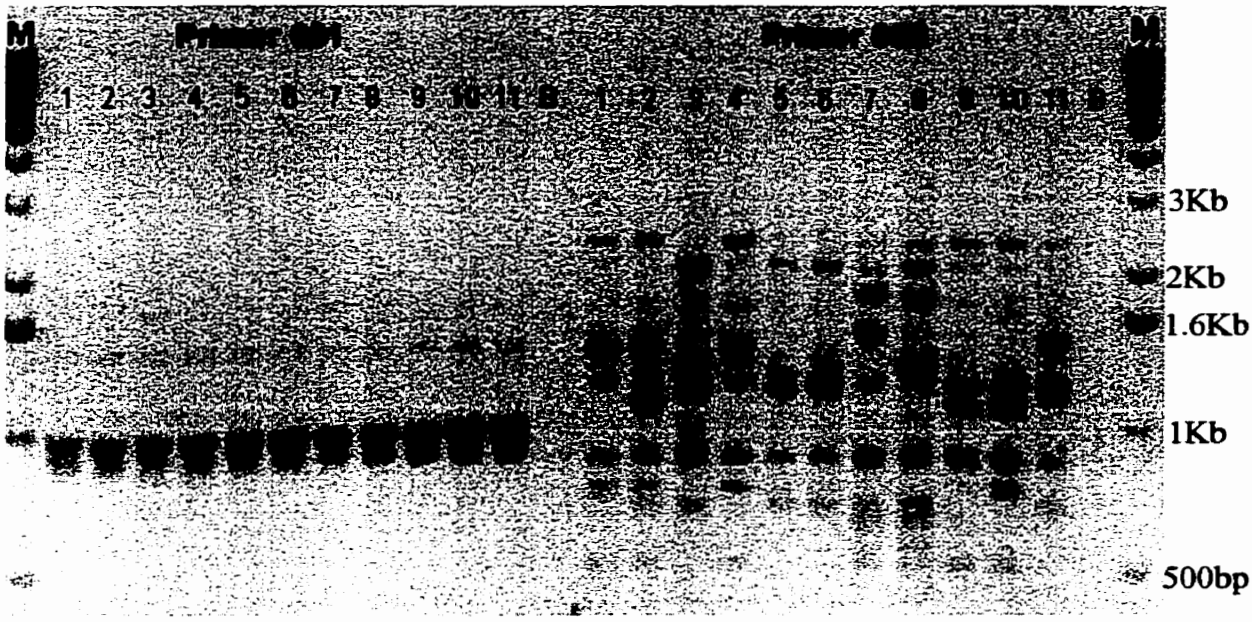


Fig 3. DNA amplification bands, obtained with primers 651 and 652 on 11 isolates from of *P. tritici-repentis*. Lanes marked M are 1-Kbp ladder DNA and those marked B are control with sterile water. Lane 1, isolate ASCI (race 1); Lane 2, isolate 94-12 (race 1); Lane 3, isolate 85-3 (race 3); Lane 4, isolate 91-18 (race 2); Lane 5, isolate 331-9 (race 3); Lane 6, isolate 331-2 (race 3); Lane 7, isolate 90-2 (race 4); Lane 8, isolate 49JA (race 4); Lane 9, isolate Alg3-24 (race 5); Lane 10, isolate Alg4-X1 (race 5); Lane 11, AlgH1 (race 6).

Note: Lack of polymorphism for primer 651 and presence of polymorphism for primer 652.

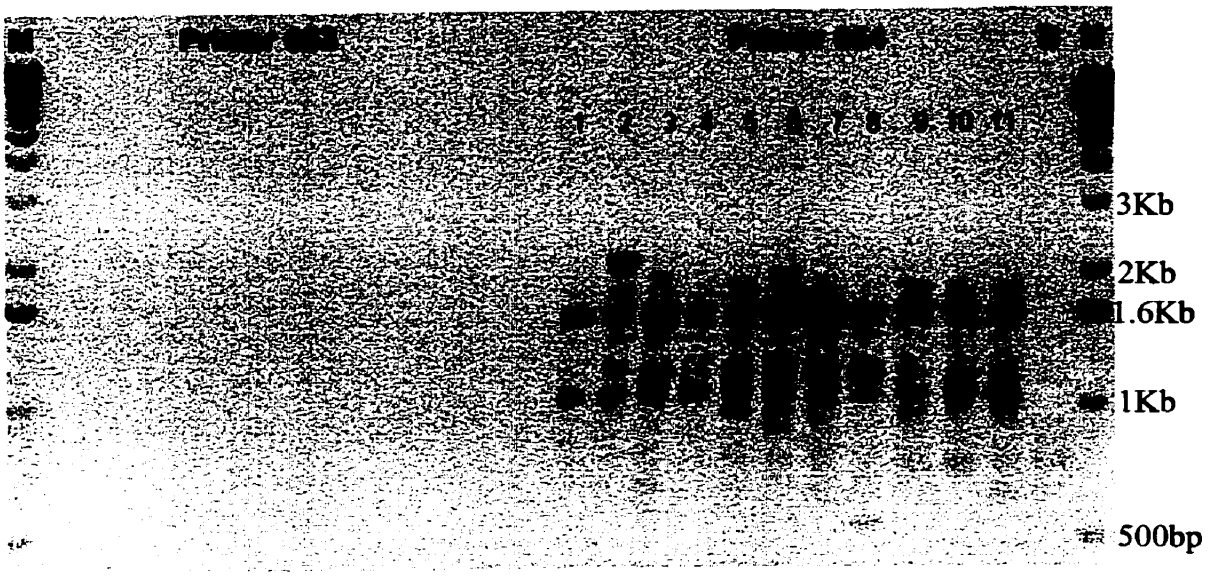


Fig 4. Screening of primers 653 and 654 with 11 isolates of *P. tritici-repentis*. Lanes marked M are 1-Kbp ladder DNA and those marked B are control with sterile water. Lane 1, isolate ASCI (race 1); Lane 2, isolate 94-12 (race 1); Lane 3, isolate 85-3 (race 3); Lane 4, isolate 91-18 (race 2); Lane 5, isolate 331-9 (race 3); Lane 6, isolate 331-2 (race 3); Lane 7, isolate 90-2 (race 4); Lane 8, isolate 49JA (race 4); Lane 9, isolate Alg3-24 (race 5); Lane 10, isolate Alg4-X1 (race 5); Lane 11, AlgH1 (race 6).
Note: No amplification products for primer 653 and presence of polymorphic fragments for primer 654.

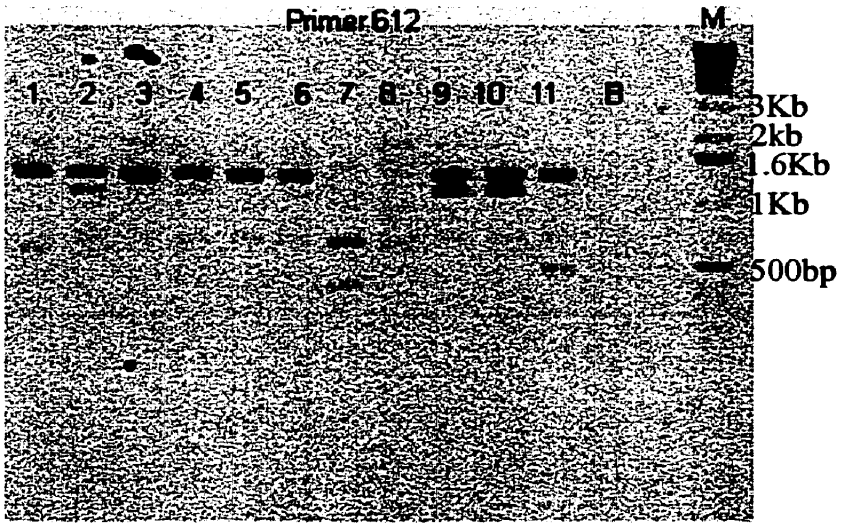


Fig 5. DNA amplification bands obtained with primer 612 on 11 isolates of *P. tritici-repentis*. Lanes marked M are 1-Kbp ladder DNA and those marked B are control with sterile water.
Lane 1, isolate ASCI (race 1); Lane 2, isolate 94-12 (race 1); Lane 3, isolate 85-3 (race 3); Lane 4, isolate 91-18 (race 2); Lane 5, isolate 331-9 (race 3); Lane 6, isolate 331-2 (race 3); Lane 7, isolate 90-2 (race 4); Lane 8, isolate 49JA (race 4); Lane 9, isolate Alg3-24 (race 5); Lane 10, isolate Alg4-X1 (race 5); Lane 11, AlgH1 (race 6).
Note: Presence of some polymorphism for primer 612 in initial screening.

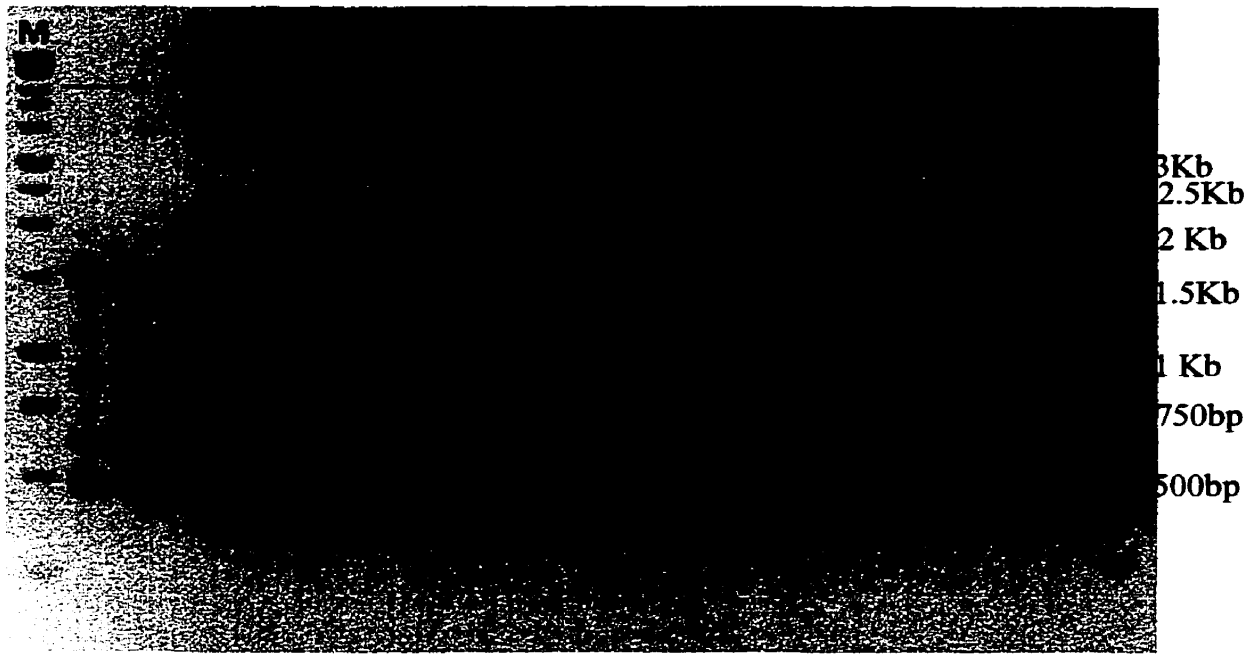
Table 3. Number of scored polymorphic RAPD bands produced by *Pyrenophora tritici-repentis*, using 8 primer from primer set #600 from the department of Microbiology, UBC, B. C.

Primer	Nucleotide sequence	Polymorphic bands
601	5'-CCGCCCACTG-3'	3
602	5'-GCGAAGACTA-3'	11
609	5'-ACAGCACCAT-3'	6
638	5'-GCGGTGACTA-3'	2
648	5'-GCACGCGAGA-3'	2
649	5'-AATGCTGGAC-3'	7
650	5'-AGTATGCAGC-3'	8
652	5'-CCCAACACAC-3'	18
Total		59

Polymorphism was found among isolates within races and among races of *P. tritici-repentis* in this study (Fig. 6, 7). The presence of polymorphism among races was found in rust fungi, *Puccinia striiformis* f. sp. *tritici* (Chen et al., 1993) and *Puccinia recondita* f. sp. *tritici* (Kolmer et al.,1995). A high degree of molecular polymorphism was found among isolates of *P. striiformis* f. sp. *tritici* that had the same virulence phenotype (Chen et al. 1993). Chen et al. (1993) concluded that the molecular polymorphism observed in *P. striiformis* was largely independent of host selection for virulence polymorphism. Kolmer et al. (1995) examined RAPD polymorphism within and between races of *Puccinia recondita* f. sp. *tritici* collected from wheat in the eastern Prairies and Pacific regions of Canada. They found that although molecular variation was greatest between isolates of different virulence phenotypes, there was some molecular variation among isolates with identical virulence.

Based on RAPD patterns, none of the 59 polymorphic markers was found to be specific to a particular race or pathotype. Specific RAPD markers or patterns for pathotypes or species are not commonly found in studies of phytopathogenic fungi. Shi et al. (1996) used RAPD markers to study the genetic relationship among wheat bunt species *Tilletia tritici, T. laevis* and *T. controversa*. They found that none of the RAPD polymorphic markers gave a clear separation at the species level because of an overlapping of the markers among the species. They also found that two species of common bunt fungi (*T. tritici* and *T. laevis*) had more genetic similarities between them than with the dwarf bunt fungus *T. controversa*. Molecular markers such as AFLP, RFLP and RAPD are generally assumed to be independent of host selection. Since RAPD markers are amplification products of

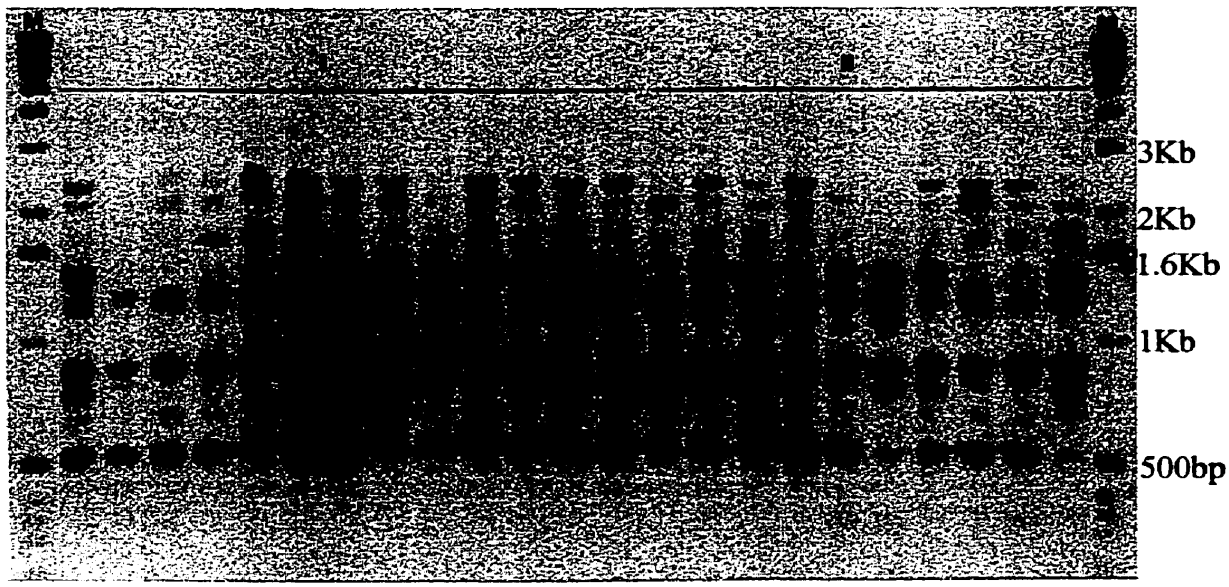
A



B

Fig 6. Agarose electrophoresis gel showing random amplified polymorphic DNA (RAPD) banding patterns of *P. tritici-repentis* amplified with primer 650. Lanes marked M are 1-Kbp DNA ladder. **A**, Thirteen isolates of race 1 (I) and nine isolates of race 2 (II). **B**, five isolates of race 2 (II), six isolates of race 3 (III), two isolates of race 4 (IV), six isolates of race 5 (V), two isolates of race 6 (VI), one isolate of race 1 (I) and control with sterile water (B).

A



B

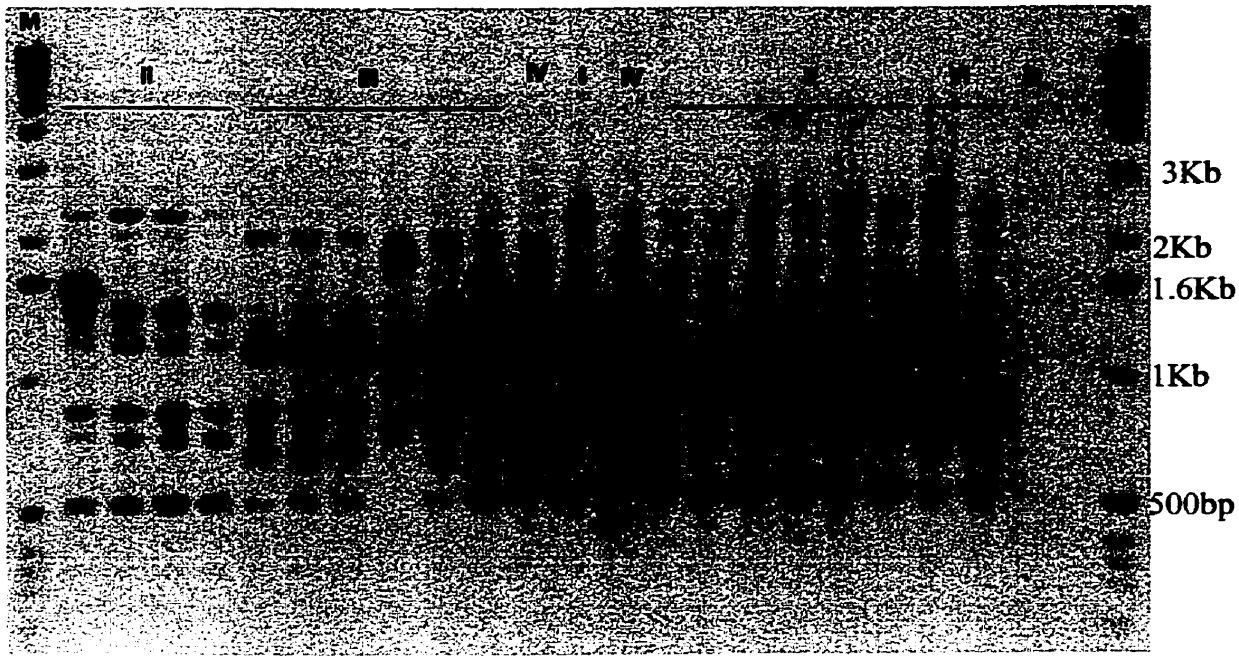


Fig 7. Agarose electrophoresis gel showing the random amplified polymorphic DNA (RAPD) banding patterns of *P. tritici-repentis* amplified with primer 652. Lanes marked M are 1-Kbp DNA ladder. **A**, Thirteen isolates of race 1 (I) and nine isolates of race 2 (II). **B**, five isolates of race 2 (II), six isolates of race 3 (III), two isolates of race 4 (IV), six isolates of race 5 (V), two isolates of race 6 (VI), one isolate of race 1 (I) and control with sterile water (B).
Note: The complex polymorphic pattern produced with this primer.

repetitive sequences that are scattered in the genome, it is not surprising that we did not find any RAPD marker or marker combination specific to a particular race or pathotype. However, necrosis-inducing races 1 and 2 isolates of *P. tritici-repentis* have similar RAPD markers although they were sampled over a 15-year period from different hosts (tetraploid, hexaploid) and geographic locations. For example, races 1 and 2 isolates obtained from tetraploid wheat in Saskatchewan shared more RAPD markers with race 1 isolate sampled from hexaploid wheat in Manitoba than with race 3 isolates sampled from tetraploid wheat in Saskatchewan (Fig. 8).

4.2 Phylogenetic Analysis of RAPD Data

4.2.1 UPGMA Analysis

Based on UPGMA analysis, 53 isolates of *P. tritici-repentis* were clustered as two distinct RAPD groups (Gp I and Gp II) (Fig. 9). Gp I contained all 51 virulent isolates (races 1, 2, 3, 5, 6) and Gp II included the only two avirulent isolates (race 4) available to us. Avirulent isolates are rarely obtained from field collections, due to the fact that all isolations are made from sporulating lesions on non-senescent leaves, a procedure which clearly excludes avirulent isolates (Lamari and Bernier 1989b). It is unlikely that the small sample size of race 4 (2 isolates) affected significantly the outcome of UPGMA. This is justified on the following grounds: i) the two isolates of race 4 were obtained in two different years (1990 and 1993) from two different geographic locations (Manitoba and Alberta), more than 1500 km apart, making it highly improbable for those isolates to be of similar genotype, ii) the two isolates do not have an identical RAPD pattern (i.e are not clones) and iii) race 6,

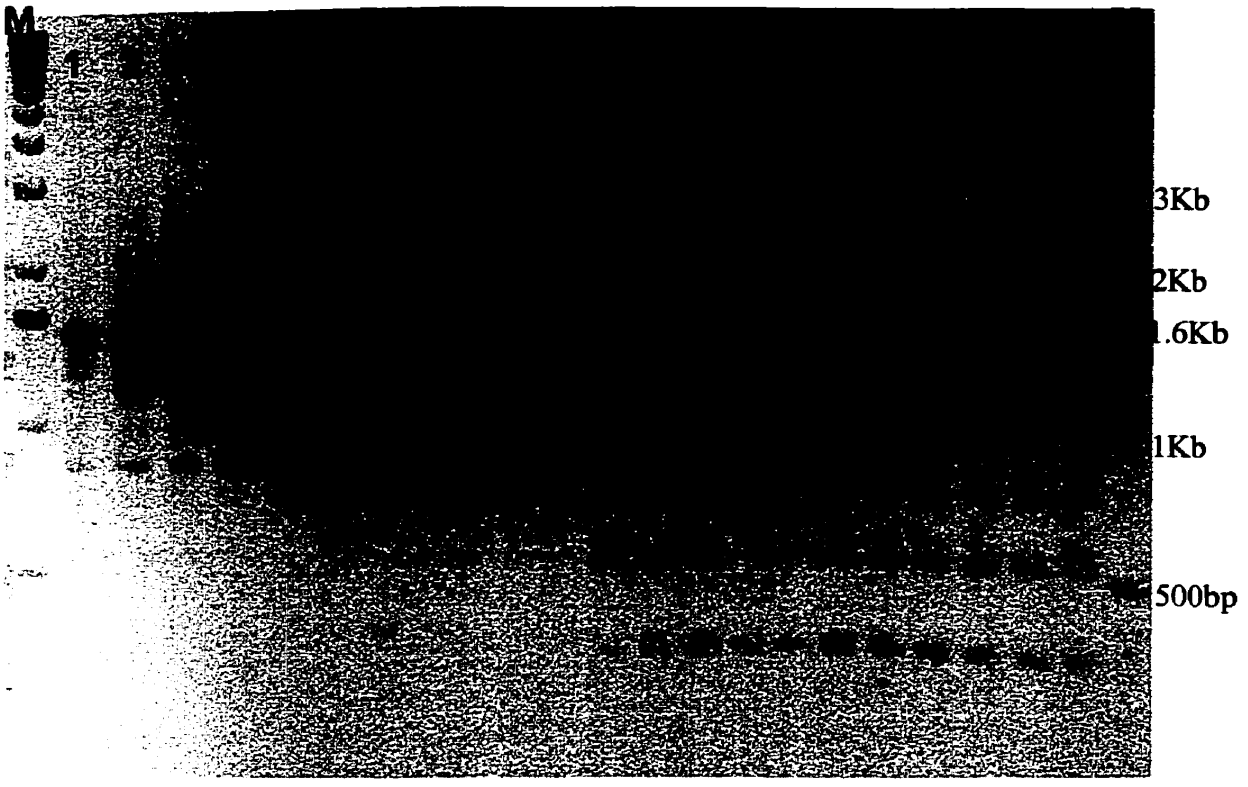


Fig 8. RAPD banding pattern of *P.tritici-repentis* isolates from Saskatchewan and one isolate from Manitoba with primer 601 and 602. Lanes marked M are 1-Kbp ladder DNA. Lane 1, isolate SC24-3 (race 1, Saskatchewan); Lane 2, SC29-2 (race 1, Saskatchewan); Lane 3, SC10-1 (race 1, Saskatchewan); Lane 4, SC3-11 (race 2, Saskatchewan); Lane 5, SC3-4 (race 2, Saskatchewan); Lane 6, SC24-2 (race 2, Saskatchewan); Lane 7, SC22-2 (race 3, Saskatchewan); Lane 8, SC29-1 (race 3, Saskatchewan); Lane 9, SC29-9 (race 3, Saskatchewan); SC 29-8 (race 3, Saskatchewan); Lane 10, ASCI (race 1, Manitoba). All Saskatchewan isolates were sampled from durum wheat and the Manitoba isolate was sampled from hexaploid wheat.
Note: The similar RAPD pattern between isolates of races 1, 2 from Saskatchewan and race 1 isolate from Manitoba.

which consisted of only two isolates, was correctly clustered with virulent races, closest to race 5 with which it shares virulence factors and geographic origin (Lamari et al., 1995). Hence, the two groups (Gp I and Gp II) are believed to be distinct.

Gp I was further divided around a 49% similarity level into two subgroups: Gp IA and Gp IB. Gp IA is composed of all necrosis-inducing isolates (nec^+) (races 1 and 2) with the exception of isolate 94-115 (race 3) and Gp IB is made up of all non-necrosis-inducing isolates (nec^-) (races 3, 5, 6) (Fig. 9). The separation between nec^+ isolates and nec^- isolates by UPGMA agreed with the previous classifications of virulent isolates based on specific induction of necrosis and chlorosis in selected host differential lines (Lamari and Bernier 1989b and Lamari et al., 1995). The necrosis inducing races 1 and 2 in Gp IA possess the highly conserved Ptr-ToxA encoding gene, responsible for inducing necrosis in the corresponding wheat differential host, Katepwa and Glenlea (Ballance et al., 1996). The nec^-, chlorosis-inducing isolates in Gp IB were recovered from tetraploid wheats in North America (race 3) and Eastern Algeria (races 5 and 6). These races were shown to carry functional (races 5 and 6), and non-functional (race 3) copie(s) of the gene encoding the host-specific chlorosis toxin, Ptr-ToxB (G M Ballance, unpublished data). Race 1 isolates also possess a chlorosis inducing factor which is shared with races 3 and 6 isolates and cause chlorosis in line 6B365. This chlorosis inducing factor in race 1 does not seemed to have an important role in genetic similarities and grouping of *P. tritici-repentis* isolates in this study.

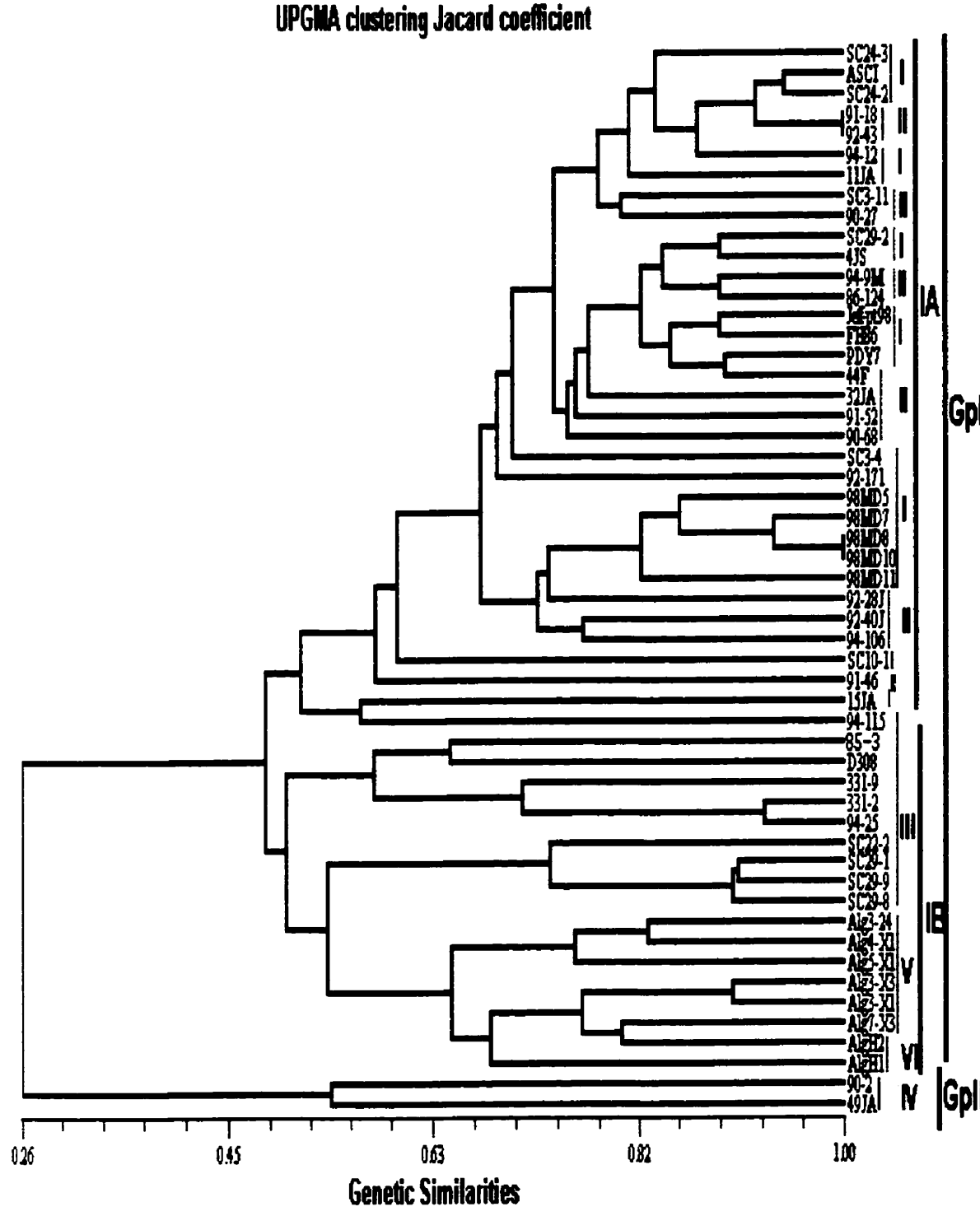


Fig 9. UPGMA clustering of 53 *P. tritici-repentis* isolates based on 59 RAPD bands. The UPGMA method in the SAHN program of NTSYSpc (version 2.0) was used. I = race 1, II = race 2, III = race 3, IV = race 4, V = race 5 and VI = race 6.

4.2.2 Neighbor Joining Distance Method

The clustering obtained by UPGMA was essentially reproduced by the neighbor joining distance method. The UPGMA works well on the assumption that the rate of substitutions is constant and the distances among the taxa are large, assumptions which are not reasonable for many data sets (Saitou and Nei, 1987). By contrast, the assumption of a constant rate of substitution is not required for neighbor joining analysis, and simulations suggest that neighbor joining is better at obtaining the correct phylogeny (Saitou and Nei, 1987). The algorithm of UPGMA iteratively joins the two nearest clusters (or groups of species), until one cluster is left (Michener and Sokal, 1957), whereas the concept of neighbor joining is to join clusters that are not only close to one another, but are also far from the rest. In each iteration, the algorithm attempts to find the direct ancestor of two species in the tree. Neighbor-joining method does not require that all lineages have diverged by equal amounts, thus, the method is especially suited for data sets comprising lineages with largely varying rates of evolution. Consider a situation where isolates A, B and C diverged from a common ancestor into two different branches: A, B in one branch and C in the second. Isolates A and B again diverged from one another into isolate A in one branch and isolate B in the other, but B has accumulated mutations at a higher rate than A and C. UPGMA method would group together A and C rather than A and B. In such a case the neighbor joining method would be the preferred procedure.

There was a change in rearrangement of some race 1 and 2 isolates from Saskatchewan and Manitoba in neighbor joining distance tree. Using UPGMA analysis, isolates SC24-2 (race 2) from Saskatchewan grouped with isolate ASCI (race 1) from

Manitoba and Saskatchewan isolate SC3-11 (race 2) was found to group with Manitoba isolate 90-27 (race 2) (Fig. 9). However in neighbor joining distance tree, isolates SC24-2 (isolate #49), SC3-11 (isolate #47) were found to be grouped together along with isolates SC3-4 (isolate #48, race 2), SC24-3 (isolate #44, race 1) from Saskatchewan (Fig. 10). Manitoba isolates ASCI (isolate #1), 90-27 (isolate #20) were also found to cluster with other races 1 and 2 isolates {98-MD11(isolate #6), 92-43 (isolate #25), 91-18 (isolate #15)} from Manitoba in neighbor joining tree (Fig. 10). Clustering of these four Saskatchewan isolates together rather than grouping with Manitoba isolates makes better sense since these four SC isolates were sampled from the same geographic location and on the same ploidy host (durum wheats). Naturally, these isolates should be more genetically similar than isolates from different hosts and different locations.

Despite minor discrepancies between UPGMA and neighbor joining methods in grouping of isolates, the general topologies of both trees were virtually the same. In neighbor joining phylogenetic analysis with our RAPD data, all the isolates of Gp IA (nec^+ races), identified by UPGMA analysis, clustered as one group. However, the neighbor joining distance tree gave clear subdivision of isolates from Gp IB (nec^- races) (Fig. 10). Race 3 isolates from Manitoba and Saskatchewan formed two separate sub-groups (Fig. 10) in neighbor joining distance tree, suggesting that geographic origin may play a role in genetic variation of race 3 isolates. The race 3 isolates from Saskatchewan were sampled not only from a single field but also at the same growing season. Thus, the year of collection could have influenced the closed grouping of these Saskatchewan isolates. González et al. 1998 found that Mexican isolates of *Colletotrichum lindemuthianum* were grouped in relation to

their geographical locations, but found no direct correlation between molecular genotype and pathotype. They concluded that the differences in the pathogen populations between regions reflected the differences in germ plasm used and the agricultural practices employed in each region. Isolates of race 3 in our study may have originated from multiple molecular backgrounds and probably evolved on different wheat cultivars in two different geographical locations. Isolates of race 5 formed a sub-group with two isolates of race 6 (Fig. 10). The isolates from race 5 and race 6 are expected to have very similar genetic background since they were all isolated from durum wheats from eastern Algeria and they both cause chlorosis on cultivar Katepwa (Ptr ToxB). Two isolates of race 4 (avirulent race) formed a distinct group by themselves. In a pathogen with sexual reproduction, every sexual generation can produce a series of individual nuclei which are different from one another. However, multiplication and dispersal will require an asexual phase of reproduction and it is during the asexual phase of reproduction that most of the increase of the virulent strains of the pathogen occurs. Avirulent isolates of *P. tritici-repentis* with a molecular phenotype different from those of virulent isolates may have arisen through the sexual recombination in the pathogen. The avirulent isolates are not likely to be selected by the host and would, therefore, remain at a very low frequency in the population, making their isolation rare.

4.2.3 Polymorphism Parsimony Method

Although the neighbor joining distance method has certain advantages over UPGMA such as permitting lineages with largely different branch lengths, it also has some disadvantages. Both UPGMA and the neighbor joining distance method reduce the data to

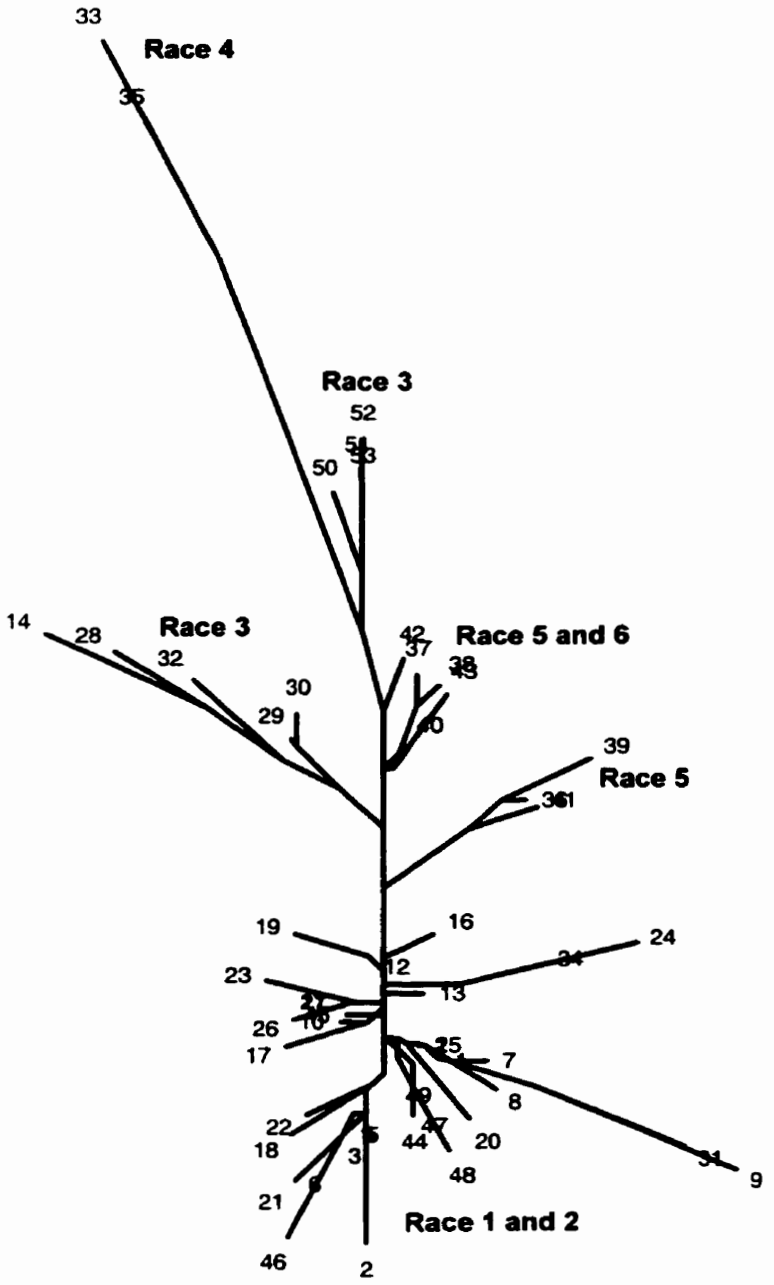


Fig 10. Radial dendrogram of 53 isolates of *P. tritici-repentis* based on 59 selected RAPD loci, using neighbor joining distance method produced by PHYLIP 3.5. The numbers refer to the isolates number from table 1.

pairwise distances, which consequently reduces the sequence information; information on the evolutionary history of the individual sites is lost (Holsinger and Jansen, 1993). The parsimony method takes this disadvantage into consideration by retaining all sequence information throughout the analysis. In a survey of 57 taxa from the sunflower family using RFLP markers, Jansen et al. (1990) found that 186 sites had only a single mutation, whereas six sites showed nine or more mutations. In such a case of different rates of substitution in the sequence data, parsimony analysis gave more reliable phylogenetic relationship data than the distance methods (Holsinger and Jansen, 1993). The parsimony method selects the tree(s) that requires the minimum amount of evolutionary change. In using the parsimony tree method, the general trend in grouping of Gp IA and sub-groupings of Gp IIA is retained except for isolates 15JA, 94-115 and Alg HI (Fig. 11). The first two isolates, 15 JA and 94-115 did not cluster with other isolates and clustered together. Isolate 94-115 (race 3) clustered with races 1 and 2 isolates as an exception in UPGMA and the neighbor joining distance tree. According to parsimony analysis, these two isolates have dissimilar molecular backgrounds and appear as outliers. Race 6 isolate, Alg HI, which grouped with the second race 6 isolate and several race 5 isolates in previous analysis was found to cluster with two race 4 isolates in parsimony analysis.

The polymorphism parsimony analysis of RAPD data (Fig. 11) largely confirmed the groupings of isolates identified by UPGMA and the neighbor joining distance methods. Like previous analyses (UPGMA, neighbor joining), the polymorphism parsimony method revealed that isolates of races 1 and 2 were closely related and there seemed to be no distinction between isolates of races 1 and 2 . Bootstrap analysis supported the clustering

of races 1 and 2 isolates as one group 30% of the time (Fig. 11) which means that according to the bootstrapping re-sampling procedure, the other arrangements in clustering of nec^+ isolates, such as grouping with nec^- isolates, occur less than 30% of the time. Bootstrapping procedure also confirmed that the arrangement of isolates in parsimony tree (Fig. 11) is the most probable arrangement out of 100 other arrangements. Isolates from race 3 formed two separate sub-groups: one sub-group consisted of isolates from Manitoba and the second composed of isolates from Saskatchewan, a result similar to the one obtained by neighbor joining method. All six isolates of race 5 and one race 6 isolate formed a sub-group in the parsimony tree as expected. The three phylogenetic analyses suggested that virulence was a major cause in genetic variation and grouping of these isolates into clusters of nec^+ and nec^- races.

The grouping of isolates based on virulence has been observed in many studies of phytopathogenic fungi. Kolmer et al. (1995) distinguished two major clusters of *Puccinia recondita* f. sp. *tritici* isolates with combined analysis of virulence and RAPD data sets. The first cluster consisted of isolates virulent or avirulent to both resistance genes *Lr2a* and *Lr2c*, and the second cluster consisted of isolates avirulent to *Lr2a* and virulent to *Lr2c*. In a study of the genetic relationships among the wheat bunt fungi, RAPD markers with bootstrap analysis supported the separation between the wheat bunt (*T. tritici* and *T. laevis*), virulent on wheat and grass bunt isolates (*T. Fusca* var. *bromi-tectorum*) fungi, virulent on cheatgrass (Shi et al., 1996). The authors also found that *T. tritici* and *T. laevis*, both virulent on wheat, were genetically very similar and suggested that these two wheat bunt fungi descended from a common ancestral population that subsequently differentiated into two sublineages. Chen

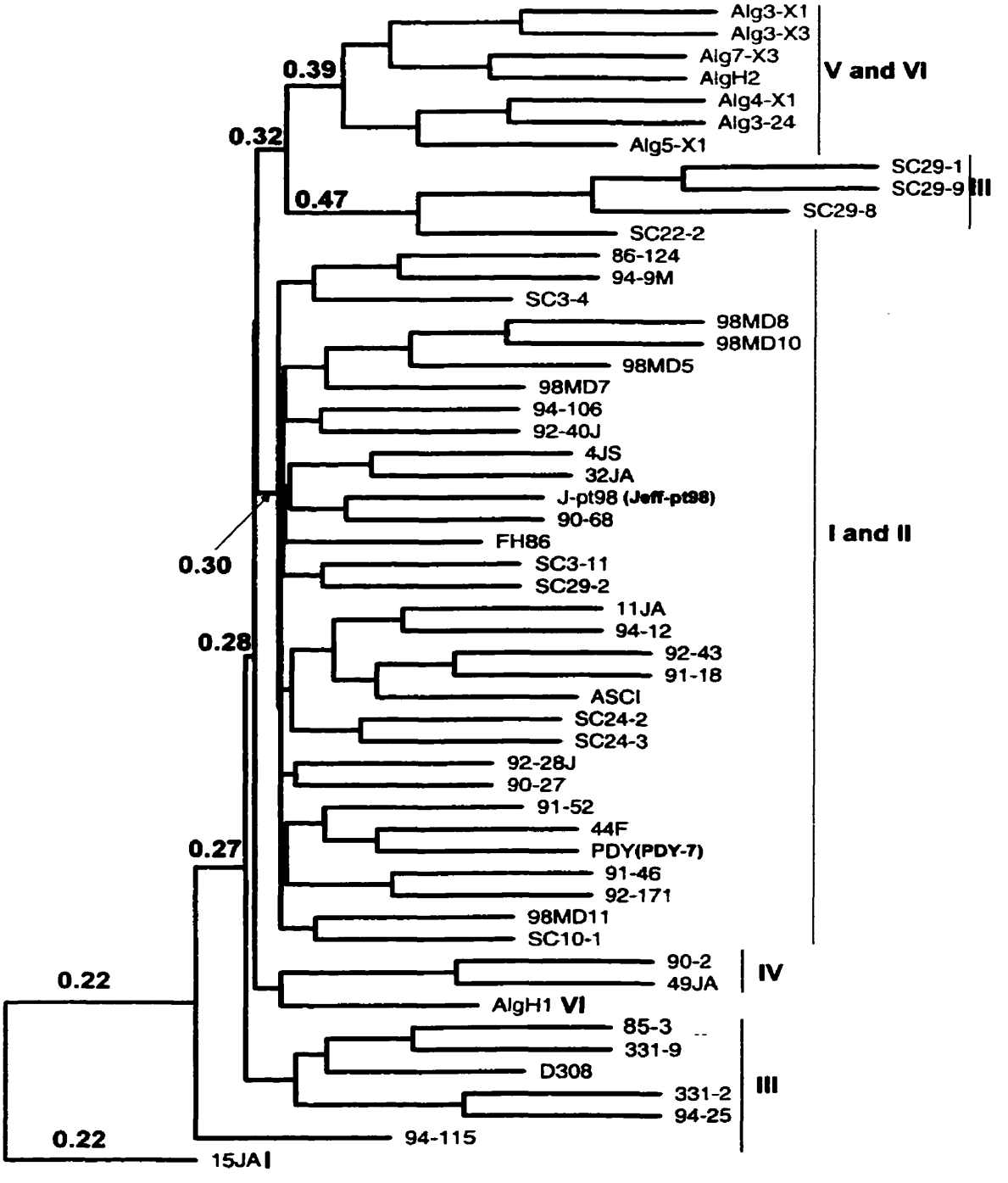


Fig 11. Dendrogram showing genetic relationships of *P. tritici-repentis* isolates studied. The polymorphism parsimony bootstrap majority-rule consensus tree was generated by PHYLIP 3.5. I= race 1, II= race 2, III= race 3, IV= race 4, V= race 5 and VI= race 6. The numbers on the branches represent the confidence intervals generated by bootstrapping with 100 replications.

et al. (1993) also found that, based on RAPD data, isolates of race CDL-21 of *Puccinia striiformis* f. sp. *tritici* formed a homogenous and distinct group from other races. Race CDL-21 is the only race detected in North America that is avirulent on cultivar Lemhi. Levy et al. (1993) examined collections of an asexual population of *Magnaporthe grisea* from a heavily infected rice nursery in Columbia for variation in DNA fingerprint and virulence on differential rice cultivars. They found that *M. grisea* isolates related on the basis of molecular variation, were also closely related for virulence.

All three phylogenetic analyses used in this study revealed similar genetic relationships among isolates of *P. tritici-repentis* such as the close relation among isolates of races 1 and 2 and distinct groupings of race 3, 5 and 6 isolates. There were a few isolates included in this study sampled from a single field in Manitoba and Saskatchewan. In most cases, isolates collected from a single field assumed to have originated from a small number of molecular backgrounds and they are expected to have very similar molecular fingerprints. The evidence of genetically identical isolates (clones) distributed in a single field was found in RFLP analysis of *Mycosphaerella graminicola* (McDonald et al., 1995) and *Sclerotinia sclerotiorum* (Kohli et al., 1992). Field epidemics of *P. tritici-repentis* are believed to be initiated by ascospores followed by successive clonal conidia production on diseased leaf tissue. It is examplified in the present study by five isolates obtained in 1998 from a single field of hexaploid wheat in the Morden (MB) region (98MD5, 98MD7, 98MD8, 98MD10, 98MD11) and three isolates obtained in 1999 from a single durum wheat field in Swift-Current (SK) region (SC29-1, SC29-9, SC29-8). The 98MD isolates were found to be genetically close and two isolates (98MD8, 98MD11) appeared to be clonal. These two are

believed to have originated from a single parental line. The Swift-Current race 3 isolates were also genetically very close, but not identical, as determined by UPGMA, neighbor-joining and parsimony analyses (Fig. 9, 10, 11). The small differences in RAPD patterns of these isolates could be due to random mutations in their genomes.

RAPD analysis suggested that isolates 91-18 and 92-43 were clonal. These two isolates were found to possess identical DNA fingerprints even though they were collected in different years and from two different fields in Manitoba. This is not an unusual case with homothallic sexually reproducing fungi, such as *P. tritici-repentis*, since homothallic sexual reproduction reduces the opportunity for genetic recombination (Kohli et al., 1992). Kohli et al. (1992) identified *Sclerotinia sclerotiorum* isolates with identical RFLP fingerprints from Manitoba and Saskatchewan in 1990 and Ontario in 1989. Clonal distributions of fungal pathogens over long distances was also observed in a study with *Mycosphaerella graminicola* (anamorph *Septoria tritici*) (Chen and McDonald 1996). The detection of *P. tritici-repentis* clones in two distant fields could be explained by host-infection and subsequent dissemination of the pathogen. Tan spot of wheat is mainly initiated by airborne ascospores in each spring. Conidia are produced asexually on the diseased leaf tissue and disseminated by wind. Schilder and Bergstrom (1992) suggested that although ascospores can be detected only 11 m away from the inoculum source, conidia can disperse more than 100 m away from the source. Presumably, clonal conidia were dispersed by wind over long distances. The identical molecular background was likely maintained through homothallic sexual reproduction and conidia were produced asexually from infected leaf tissue, initiated by a single ascospore. Identical fingerprints of 91-18 and 92-43 could have resulted from a

long distance dispersal of asexually produced conidia and lack of genetic recombination in homothallic sexual reproduction.

There is an occurrence of differential interactions between host and pathogen genotypes in many pathosystems. Since the pathogen evolved in parallel with the host, genetic variation in the pathogen population is likely to be influenced by the host. Genetic variation for virulence and resistance in the wheat-*Mycosphaerella graminicola* pathosystem was studied by Kema et al. (1996) with isolates that originated from common and durum wheat. They found that isolates of *M. graminicola* were specialized to either bread wheat or durum wheat and could be easily recognized in inoculation experiments, but could not be distinguished by amplification and digestion of nuclear and mitochondrial internally transcribed spacer ribosomal DNA (ITS rDNA). A similar situation was described for wheat leaf rust, in which the durum wheat and bread wheat types clearly differed in pathogenicity and also appeared to be sexually isolated, but could not be distinguished by molecular markers (Ezzahiri et al., 1994, Zambino and Szabo, 1993).

The toxins of *P. tritici-repentis* show selective host-specificity by causing either necrosis/ chlorosis or both symptoms in differential wheat genotypes (Lamari and Bernier, 1989c) . The role of the host (hexaploid vs tetraploid) on the evolution of *P. tritici-repentis* remains one of the most intriguing questions (Lamari et al. 1998, Gamba & Lamari 1998). To date, all isolates of races 3, 5 and 6 were recovered from tetraploid wheats, whereas races 1 and 2 were regularly found on both hexaploid and tetraploid wheats. We purposely included in this study a set of 10 isolates from races 1,2 and 3, sampled in 1999-2000 from Swift Current, the durum wheat growing region of Saskatchewan. We hypothesized that if

the host ploidy determined the specialization of the pathogen, all the 10 isolates recovered from tetraploid wheats should cluster together, irrespective of the race to which they belonged. This hypothesis was rejected as the isolates split on the basis of virulence. Isolates of races 1 and 2 from Swift Current (SC24-3, SC29-2, SC10-1, SC3-11, SC3-4, SC24-2) clustered with necrosis-inducing races, whereas isolates of race 3 (SC22-2, SC29-1, SC29-9, SC29-8) clustered with chlorosis-inducing isolates (Fig. 7, 8 and 9). Based on our results from phylogenetic analyses, virulence appears to be the single most important factor in the evolution of *P. tritici-repentis*.

4.3 Analysis of Molecular Variance (AMOVA)

AMOVA is similar to a hierarchical analysis of variance such as ANOVA in that it separates and tests tiers of genetic diversity among groups of populations, diversity among the populations within groups and diversity among the individuals within a population (Excoffier et al., 1992). AMOVA differs from conventional analysis of variance in that AMOVA can accommodate different evolutionary assumptions without modifying the basic structure of the analysis and the method is flexible enough to accommodate several alternative input matrices, corresponding to different types of molecular data (Excoffier et al., 1992). The significance of the variance components and Φ-statistics (analogous to F-statistics) is tested using a permutational method so that the assumption of random sampling and normal distribution which is inappropriate for molecular data is not required (Excoffier et al., 1992).

The partitioning of molecular variance among necrosis-inducing races (nec^+) and non-necrosis (nec^-) but chlorosis inducing races was obtained using AMOVA. The molecular variance between virulent and avirulent races was not calculated because of the small sample size of race 4 (2 isolates). From the Euclidean distance matrix among all pairs of the 51 virulent isolates, AMOVA evaluated the variance components of virulent races in two ways. When the five races are lumped into 2 groups (races 1 and 2 vs. races 3, 5, 6), differences between groups accounted for 36% of the total variance and differences between isolates within groups accounted for 64% (Table 4). The result indicated that these two groups are distinct from one another at $P< 0.001$ level. A nested analysis of variance components revealed that 31% of the variation was attributable to differences among groups (nec^+ isolates vs. nec^- isolates), 11% to differences among races withing groups and 58% to differences among individual isolates within races (Table 5). All variance components were highly significant ($P< 0.001$), demonstrating the existence of heterogeneity. The large amount of variability within races (58%) is not surprising given that the population used in this study originated from different continents, hosts and collection years. However, this variability could also reflect the fact that *P. tritici-repentis* undergoes sexual reproduction in its life cycle. Sexual reproduction was found to play a major role in the genetic structure of *M. graminicola* population (Chen and McDonald, 1996). RFLP markers showed extensive genetic variation in *M. graminicola* (McDonald and Martinez, 1990) and the finding of a high degree of genotypic diversity suggested that sexual reproduction (random mating) plays a major role in genetic structure of populations of *M. graminicola* (Chen and McDonald, 1996).

Table 4. Analysis of molecular variance (AMOVA) for 51 isolates of *P. tritici-repentis*. AMOVA measured variance among groups (races 1, 2 vs. races 3, 5 and 6).

Source of Variation	Variance component[a]	Φ statistic	P[b]
Among groups	2.1185 (35.9%)	$\Phi_{ST} = 0.359$	< 0.001
Within groups	3.7824 (64.10%)	---	---

[a]Percentage of the total variance is given in parentheses.
[b]Probability of having a more extreme variance component and Φ-statistic than the observed values by chance alone, computed by nonparametric procedure from 1000 random permutations.
—These values are not calculated by AMOVA since analysis is only for variance between groups (2-level analysis), in this case, among nec^+ races and nec^- races.

Table 5. Analysis of molecular variance (AMOVA) for 51 isolates of *P. tritici-repentis*. AMOVA measured variance among groups (races 1, 2 vs. races 3, 5 and 6), among races within two groups and among isolates within 6 races.

Source of Variation	Variance component[a]	Φ statistic	P[b]
Groups	1.8423 (31.18%)	$\Phi_{CT} = 0.312$	< 0.001
Races within groups	0.6236 (10.55%)	$\Phi_{SC} = 0.153$	< 0.001
Isolates within races	3.4425 (58.26%)	$\Phi_{ST} = 0.417$	< 0.001

[a]Percentage of the total variance is given in parentheses.
[b]Probability of having a more extreme variance component and Φ-statistic than the observed values by chance alone, computed by nonparametric procedure from 1000 random permutations.

Pairwise population comparisons examined with AMOVA resulted in values of Φ_{ST} that are interpreted as the average interpopulation distance between any two races (Race 4 and 6 isolates were excluded from the analysis due to their small sample sizes) (Table 6). The interpopulation distance between race 1 and race 2 was non-significant (P= 0.3886), supporting the results of phylogenetic analyses that these two races are extremely close. The interpopulation distances among nec$^+$ races and nec$^-$ races (between races 1 and 3, races 1 and 5, races 2 and 3, and races 2 and 5) was around 0.4 and significant at P< 0.001 (Table 6). It was found that the distance between nec$^-$ races (3 and 5) was 0.3139 and significant at less than 0.001 level (Table 6).

Table 6. The interpopulation genetic distance between 4 races of *P. tritici-repentis* calculated by AMOVA analysis. Race 1 and race 2 each contained 17 isolates, race 3 contained 9 isolates and race 5 contained 6 isolates with 59 RAPD primers.

	Race 1	Race 2	Race 3	Race 5
Race 1	----	0.4306	0.001	0.001
Race 2	0.0123	----	0.001	0.001
Race 3	0.3993	0.4575	----	0.001
Race 5	0.3991	0.4590	0.3139	----

Below diagonal: Total molecular variation residing between population (Φ_{ST}) is a measure of interpopulational genetic distance.
Above diagonal: Significant testing of each Φ_{ST} value was calculated by nonparametric procedure from 1000 random permutation as the probability that a random Φ_{ST} value was greater than the observed value.

The coancestry coefficient (modified Φ_{ST}) was calculated by AMOVA to determine which pair of races (populations) were different or similar with respect to RAPD type. The coancestry coefficients between pairs of races 1, 2, 3 and 5 gave virtually the same results as interpopulation distance between pairs of races (Table 7). Races 4 and 6 were excluded because of their small sample size. The same results as all previous analyses were obtained: the necrosis inducing races (races 1 and 2) were homogenous with respect to RAPD types. Race 1 and race 2 were not significantly different from each other (P=0.3886), although nec$^-$ races (races 3 and 5) were significantly different from each other at P< 0.001 (Table 7). The results suggested that races 1 and 2 are from the same population, whereas races 3 and 5 are of different populations. The significant differences in coancestry coefficient between nec$^+$ and nec$^-$ races further indicated that population of nec$^+$ isolates was different from populations of nec$^-$ isolates. These results show significant heterogeneity between nec$^+$ races and nec$^-$ races, but homogeneity between nec$^+$ races.

Table 7. The modified coancestry coefficient between pairs of *P. tritici-repentis* races generated from AMOVA. Race 1 contained 17 isolates, race 2 contained 17 isolates, race 3 contained 10 isolates and race 5 contained 6 isolates.

	Race 1	Race 2	Race 3	Race 5
Race 1	----	0.3886	0.001	0.001
Race 2	0.0124	----	0.001	0.001
Race 3	0.5097	0.6116	----	0.001
Race 5	0.5093	0.6144	0.3767	----

Below diagonal: Coancesty coefficient among two races
Above diagonal: Significance testing of each coancestry coefficient was performed by a nonparametric procedure using 1000 random permutation as the percent probability that a random coancestry coefficient value was greater than the observed value.

The absence of genetic structure within necrosis inducing races, in spite of the fact that isolates from this group originated from a very wide geographic area (Uruguay, USA and three western Canadian provinces), hosts (hexaploid and tetraploid) and collection year (over 15 years), strongly suggests a recent origin for races 1 and 2. A sexually reproducing pathogen can produce a new combination of virulence genes in every sexual generation that can be selected out by the corresponding host resistance genes. Under this scenario, fungal isolates having the same combination of virulence genes may have a recent common ancestor. The ability of races 1 and 2 to induce necrosis in a host-specific manner (Lamari and Bernier 1989b, 1989c), in addition to the known susceptibility to necrosis of all major Canadian hexaploid wheat cultivars, could have provided these races a clear selective advantage. In fact, surveys conducted in 1985-1988 (Lamari and Bernier 1989b) and in

1990-1994 (Lamari et al. 1998) and later (L. Lamari, unpublished data), revealed a predominance (>95%) of races 1 and 2 in the Canadian Prairies. On the other hand, the separation between chlorosis-inducing races (races 3 and 5), which are significantly different from each other based on AMOVA, is consistent with the virulence data. Isolates of races 3 and 5 possess different toxins or mechanisms, even though they all produce chlorosis symptoms on susceptible wheat genotypes. The wheat line 6B365 and cultivar Katepwa differentiated chlorosis induction by race 5 isolates from chlorosis induction by race 3 (Lamari et al., 1995). Both phylogenetic analyses and AMOVA suggested that virulence is the major force in evolution and genetic variation in *P. tritici-repentis*.

5.0 GENERAL DISCUSSION AND CONCLUSIONS

Current knowledge of genetic variation in *P. tritici-repentis* is limited to analysis of virulence data, generated by screening isolates of the pathogen on a host differential set. Even though virulence may play an important role in the epidemiology of the pathogen, it only represents a minor portion of the genetic variability in the population. RAPD data, generated in this study, provided a reasonable starting point to address several genetic aspects of *P. tritici-repentis*, including the phylogenetic relationships among the races, the role of virulence in the evolution of the pathogen and the effect of host and geographic origin on the specialization of the pathogen.

The results of phylogenetic analyses in this study indicated that avirulent race 4 was distinct from the virulent races. UPGMA and neighbor-joining distance analysis also showed that necrosis inducing races (races 1 and 2) formed a subgroup distinct from chlorosis-only inducing races (races 3, 5 and 6). Analysis of molecular variance (AMOVA) also revealed that nec^+ and nec^- races were significantly different from each other ($P<0.001$). The division between races 1 and 2 and races 3, 5 and 6 is undoubtedly the most significant and surprising result of this study, as it reproduced the previous classification of isolates, based on specific induction of necrosis and chlorosis in selected differential lines.

Regardless of the phylogenetic analysis method used in this study, races 1 and 2 appear to represent a relatively homogenous group. The interpopulation distance and coancestry coefficient among isolates of races 1 and 2, derived by AMOVA was non-significant, suggesting that the isolates of the two races were part of the same population. The fact that both races carry the necrosis toxin gene (Ptr-ToxA) and that they are the most

prevalent races in Canada, suggests that acquisition of this gene by races 1 and 2 provided them a selective advantage over other races.

Despite the fact that RAPD markers are assumed to be neutral and encompassing the whole genome, the overall DNA patterns between race 1 and race 2 are remarkably similar to each other even though some of these isolates originated from different continents. However, DNA patterns representing races 3, 4, 5 and 6 are different from the patterns representing races 1 and 2. This result, in combination with AMOVA analysis, has led us to speculate that races 1 and 2 might belong to a sub-species which is distinct from races 3, 4, 5 and 6. An alternative explanation is that races 1 and 2 may have acquired an alien chromosome or a small alien chromosome segment which confers Ptr-ToxA encoding gene that is absent in races 3, 4, 5 and 6.

Unlike necrosis inducing races, chlorosis inducing races (races 3, 5 and 6) formed separate sub-clusters in phylogenetic analyses based on virulence, geographic origin or both. AMOVA results also demonstrated that race 3 and race 5 were of two different populations. Besides possessing different chlorosis toxins/mechanisms, the geographic origin of the races may have played a role in genetic variation of isolates. Race 5 and race 6 isolates which share the ability to cause chlorosis on Katepwa and also from the same geographic origin grouped closely together in phylogenetic analyses. Race 3 isolates from Manitoba and Saskatchewan formed separate sub-groups in both distance and parsimonous trees, although they both share the same virulence factor such as the ability to induce chlorosis on line 6B365.

The genetic variation of *P. tritici-repentis* appeared to depend on factors such as virulence, host-ploidy level and geographic origin. Host-ploidy level and geographic origin of the pathogen appeared to play a minor role in genetic variation of *P. tritici-repentis*. The results of this study suggested that virulence played a major role in specialization and evolution of the pathogen. This was evident in results from analysis of isolates of races 1 and 2 obtained from durum wheat in Saskatchewan. Isolates recovered from a single field in Saskatchewan were found to be clustered in two different groups in phylogenetic analyses on the basis of their virulence instead of clustering in one group based on their geographic origin or host. In conclusion, the results of this study identified virulence as the single most important factor in genetic variation and evolution of *P. tritici-repentis*.

This study represents a starting point for investigating the genetic variation in *P. tritici-repentis*. A similar study could be carried out on ascospores-derived isolates from pseudothecia. Recovering isolates directly from pseudothecia and not from lesions will give us a better opportunity to assess the true genetic variability generated by the sexual cycle, without the filtering effect of host-selection. It will be also interesting to examine the genetic structure of the pathogen population in a single field naturally infected with tan spot. Molecular markers can be used to measure the effect of sexual reproduction (ascospres and conidia from early season) and asexual reproduction (conidia from late season) on the genetic structure of the pathogen population over the course of an epidemic cycle.

LITERATURE CITED

Apuya, N., Frazier, B., Keim, P., Roth, E. J. and Lark, K. 1988. Restriction length polymorphisms as genetic markers in soybean *Glycine max* (L.) Merr. Theoretical and Applied Genetics 75: 889-901.

Ballance, G. M., Lamari, L., Kowatsch, R. and Bernier, C. C. 1996. Cloning, expression and occurrence of the gene encoding the Ptr necrosis toxin from *Pyrenophora tritici-repentis*. Molecular Plant Pathology On-Line, http://www.bssp.org.uk/mppol/1996/1209ballance.

Ballance, G. M., L. Lamari, and C. C. Bernier. 1989. Purification and characterization of a host selective toxin from *Pyrenophora tritici-repentis*. Physiological and Molecular Plant Pathology 35: 203-213.

Boeger, J. M., Chen, R. S., and McDonald, B. A. 1993. Gene flow between geographic populations of *Mycosphaerella graminicola* (Anamorph *Septoria tritici*) detected with restriction fragment length polymorphism markers. Phytopathology 83: 1148-1154.

Bonants, P. J. M., Hagenaar, De Veerdt, Van Gent-Pelzer, M., Lacourt, M. P., Cooke, D. E. L. and Duncan, J. M. (1997). Detection and identification of *Phytophthora fragariae* Hickman by the polymerase chain reaction. European Journal of Plant Pathology. 103: 345-355.

Borchardt, D. S., Welz, H. G. and Geiger, H. H. 1998. Genetic structure of *Setosphaeria turcica* populations in tropical and temperate climates. Phytopatholgy. 88: 322-329.

Bruns, T. D., White, T. J. and Taylor, J. W. 1991. Fungal molecular systematics. Annual Review of Ecology and Systematics 22: 525-564.

Burdon, J. J. and Roelfs, A. P. 1985. Isozyme and virulence variation in asexually reproducing populations of Puccinia graminis and *P. recondita* on wheat. Phytopathology 75: 907-913.

Carter, D. A., Archer, S. A., Buck, K. W., Shaw, D. S. and Shattock, R. C. (1990). Restriction fragment length polymorphisms of mitochondrial DNA of *Phytophthora infestans*. Mycological Research. 94: 1123-1128.

Chen, R-S., and McDonald, B. A. 1996. Sexual reproduction plays a major role in the genetic structure of populations of the fungus *Mycosphaerella graminicola*. Genetics 142: 1119-1127.

Chen, X., Line, R. F. and Leung, H. 1993. Relationship between virulence variation and DNA polymorphism in *Puccinia striiformis*. Phytopathology. 83: 1489-1497.

Ciuffetti, L. M., Tuori, R. P. and Gaventa, J. M. 1997. A single gene encodes a selective toxin caused to the development of tan spot on wheat. The Plant Cell 9: 135-144.

Coddington, A. and Gould, D. S. 1992. Use of RFLPs to identify races of fungal pathogens. In: Duncan, J. M. and Torrance, L. (eds.), Techniques for the rapid detection of plant pathogens, Blackwell Scientific Publications, Oxford, UK.

Cooke, D. E. L. and Duncan, J. M. 1997. Phylogenetic analysis of *Phytophthora* species based on ITS1 and ITS2 sequences of the ribosomal RNA gene repeat. Mycological Research 101: 667-677.

Day, J. P. and Shattock, R. C. 1997. Aggressiveness and other factors relating to displacement of populations of *Phytophthora infestans* in England and Wales. European Journal of Plant Pathology. 103: 379-391.

De Cock, A. W. A. M., Neuvel, A., Bahnweg, G., De Cock, J. C. J. M. and Prell, H. H. (1992). A comparison of morphology, pathogenicity and restriction fragment length patterns of mitochondrial DNA among isolates of *Phytophthora porri* Foister. Netherlands Journal of Plant Pathology. 98:277-289.

De Wolf, E. D., and L. J. Francl. 1998. Empirical infection period models for tan spot of wheat. Canadian Journal of Plant Pathology 20:394-395.

Dhingra, O. D., and Sinclair, J. B. 1985. Basic Plant Pathology Methods. pp. 355. CRC Press, Boca Raton, Florida..

Dowling, T. E., Moritz, C. and Palmer, J. D. 1990. Nucleic acids II: Restriction site analysis. In: Hillis, D. M. and Moritz, C. (eds.), Molecular systematics, Sinauer Associates, Inc. Publishers, Massachusetts, USA.

Duncan, J. M., Cooke, D., Birch, P. and Toth, R. 1998. Molecular variability in sexually reproducing fungal plant pathogens. In: Bridge, Y. Couteaudier, J. M. Clarkson (eds.). Molecular Variability of Fungal Pathogens. P. D. CAB international, Wallingford, UK.

Excoffier, L., Smouse, P. E., and Quattro, J. M. 1992. Analysis of molecular variance inferred from metric distances among DNA haplotypes: Application to human mitochondrial DNA restriction data. Genetics 131: 479-491.

Ezzahiri, B., Diouri, S., and Roelfs, A. P. 1994. Pathogenicity of *Puccinia recondita* f.sp. *tritici* in Morocco during 1985, 1988, 1990, and 1992. Plant Disease 78: 407-410.

Farris, J. S. 1977. Phylogenetic analysis under Dollo's law. Systematic Zoology 26: 77-88.

Farris, J. S. 1970. Methods for computing Wagner trees. Systematic Zoology 19: 83-92.

Felsenstein, J. 1993. PHYLIP (Phylogenetic Inference Package) Version 3.5 c. University of Washington, Seattle, Washington 98195.

Felsenstein, J. 1979. Alternative methods of phylogenetic interference and their interrelationship. Systematic Zoology 28: 49-62.

Förster, H. and Coffey, M. D. (1993). Molecular taxonomy of *Phytophthora megasperma* based on mitochondrial and nuclear DNA polymorphisms. Mycological Research. 97: 1101-1112.

Gamba, F. M. and Lamari, L. 1998. Mendelian inheritance of resistance to tan spot [*Pyrenophora tritici-repentis*] in selected genotypes of durum wheat (*Triticum turgidum*). Canadian Journal of Plant Pathology 20: 408-414.

González, M., Rodriguez, R., Zavala, M. E., Jacobo, J. L., Hernandez, F., Acosta, J., Martinez, O. and Simpson, J. 1998. Characterization of Mexican isolates of *Colletotrichum lindemuthianum* by using differential cultivars and molecular markers. Phytopathology 88: 292-299.

Goodwin, P. H. and Annis, S. L. 1991. Rapid identification of genetic variation and pathotype of *Leptosphaeria maculans* by random amplified polymorphic DNA assay. Applied and Environmental Microbiology 57: 2482-2486.

Grajal-Martin, M. j., Simon, C. J. and Muchlbauer, F. J. 1993. Use of random amplified polymorphic DNA (RAPD) to characterize race 2 of *Fusarium oxysporum* f. sp. *pisi*. Phytopathology 83: 612-614.

Hamelin, R. C., Dusabenyagasani, M. and Et-touil, K. 1998. Fine-level genetic structure of White pine blister rust populations. Phytopathology 88: 1187-1191.

Hedrick, P. 1992. Shooting the RAPDs. Nature 355: 679-680.

Hillis, D. M., Allard, M. W. and Miyamoto, M. M. 1993. Analysis of DNA sequence data: Phylogenetic interference. Methods in Enzymology 224: 456-487.

Hosford, R. M. Jr. 1982. Tan spot. Pages 1-24. In: Tan Spot of Wheat and Related Diseases. R. M. Jr. Hosford. ed. North Dakota State University.

Hosford, R. M., Jr. 1971. A form of *Pyrenophora trichostoma* pathogenic to wheat and other grasses. Phytopathology 61:28-32.

Hosford, R. M., Jr. 1972. Propagules of *Pyrenophora trichostoma*. Phytopathology 62:627-629.

Holsinger, K. E. and Jansen, R. K. 1993. Phylogenetic analysis of restriction site data. Methods in Enzymology. 224: 439-455.

Howard, R. J., and R. A. A. Morrall. 1975. The epidemiology of leaf spot disease in a native prairie. I. The progression of disease with time. Canadian Journal of Botany 53:1040-1050.

Huff, D. R. 1997. RAPD characterization of heterogeneous perennial ryegrass cultivars. Crop Science 37:557-564.

Huff, D. R., Peakall, R. and Smouse, P. E. 1993. RAPD variation within and among natural populations of outcrossing buffalograss Buchloe dactyloides (Nutt.) Engelm.]. Theoretical and Applied Genetics 86:927-934.

Isabel, N., Beaulieu, J. and Bousquet, J. 1995. Complete congruence between gene diversity estimates derived from genotypic data at enzyme and random amplified polymorphic DNA loci in black spruce. Proceedings of the National Academy of Science of the USA 92:6369-6373.

Jacard, P. 1908. Nouvelles recherches sur la distribution florale. Société Vaudoise des Sciences Naturelles. Bulletin 44:223-270.

Jansen, R. K., Holsinger, K. E., Michaels, H. J. and Palmer, J. D. 1990. Phylogenetic analysis of chloroplast DNA restriction site data at higher taxonomic levels: an example from the *Asteraceae*. Evolution. 44: 2089-2105.

Kelly, A., Alcala-Jimenez, A. R., Bainbridge, B. W., Heale, J. B., Perez-Artes, E. and Jimenez-Diaz, R. M. 1994. Use of genetic fingerprinting and random amplified polymorphic DNA to characterize pathotypes of *Fusarium oxysporum* f. sp. *ciceris* infecting chickpea. Phytopathology 84: 1293-1298.

Kema, G. H. J., Annone, J. G., Sayoud, R., Van Silfhout, C. H., Van Ginkel, M. V. and de Bree, J. 1996. Genetic variation for virulence and resistance in the wheat-*Mycosphaerella graminicola* pathosystem I. Interactions between pathogen isolates and host cultivars. Phytopathology 86: 200-212.

Kohli, Y., Morrall, R. A. A., Anderson, J. B., and Kohn, L. M. 1992. Local and trans-Canadian clonal distribution of *Sclerotinia sclerotiorum* on Canola. Phytopathology 82: 875-880.

Kohn, L. M. 1992. Developing new characters for fungal systematics: an experimental approach for determining the rank of resolution. Mycologia 84: 139-153.

Kohn, L. M., Stasovski, E., Carbone, I., Royer, J. and Anderson, J. B. 1991. Mycelial imcompatibility and molecular markers identify genetic variability in field populations of *Sclerotinia sclerotiorum*. Phytopathology 81: 480-485.

Kolmer, J. A., Liu, J. Q. and Sites, M. 1995. Virulence and molecular polymorphism in *Puccinia recondita* f. sp. *tritici* in Canada. Phytopathology 85: 276-285.

Krupinsky, J. M. 1992. Grass hosts of *Pyrenophora tritici-repentis*. Plant Disease 68: 13-16.

Lamari, L., Gilbert, J. and Tekauz, A. 1998. Race differentiation in *Pyrenophora tritici-repentis* and survey of physiologic variation in western Canada. Can. J. Plant Pathol. 20: 396-400.

Lamari, L., Sayoud, R., Boulif, M. and Bernier, C. C. 1995. Identification of a new race in *Pyrenophora tritici-repentis*: implications for the current pathotype classification system. Canadian Journal of Plant Pathology 17: 312-318.

Lamari, L., and Bernier, C. C. 1991. Genetics of tan necrosis and extensive chlorosis in tan spot of wheat caused by *Pyrenophora tritici-repentis*. Phytopathology 81: 1092-1095.

Lamari, L., and Bernier, C. C. 1989a. Evaluation of wheat lines and cultivars to tan spot (*Pyrenophora tritici-repentis*) based on lesion type. Canadian Journal of Plant Pathology 11: 49-56.

Lamari, L. and Bernier, C. C. 1989b. Virulence of isolates of *Pyrenophora tritici-repentis* on 11 wheat cultivars and cytology of the differential host reactions. Canadian Journal of Plant Pathology 11: 284-290.

Lamari, L. and Bernier, C. C. 1989c. Toxin of *Pyrenophora tritici-repentis*: host-specificity, significance in disease, and inheritance of host reaction. Phytopathology 79: 740-744.

Lardner, R., Johnston, P. R., Plummer, K. M. and Pearson, M. N. 1999. Morphological and molecular analysis of *Colletotrichum acutatum sensu lato*. Mycological Research 103: 275-285.

Levy, M., Correa-Victoria, F. J., Zeigler, R. S., Xu, S., and Hamer, J. E. 1993. Genetic diversity of the rice blast fungus in a disease nursery in Columbia. Phytopathology 83: 1427-1433.

Longato, S. and Bonfante, P. 1997. Molecular identification of mycorrhizal fungi by direct amplification of microsatellite regions. Mycological Research 101: 425-432.

Luttrel, S. E. 1973. Loculoascomycetes. Pages 135-219 in G. C. Ainsworth, F. K. Sparrow and A. S. Sussman, eds., The fungi, IVA. A taxonomic review with keys. Academic Press, New York, NY. 621 pp.

Majer, D., Mitten, R., Lewis, B., G., Vos, P. and Oliver, R. P. 1996. The use of AFLP fingerprinting for the detection of genetic variation in fungi. Mycological Research. 100: 1107-1111.

McCallum, B. D., Bernier, C. C. and Lamari, L. L. 1992. Generation and utilization of chemical-resistant mutants in *Pyrenophora tritici-repentis*, the causal agent of tan spot of wheat. Canadian Journal of Botany 72: 100-105.

McDonald, B. A., Pettway, R. E., Chen, R. S., Boeger, J. M. and Martinez, J. P. 1995. The population genetics of *Septoria tritici* (teleomorph *Mycosphaerella graminicola*). Canadian Journal of Botany 73(Suppl. 1): S292-S301.

McDonald, B. A. and Martinez, J. P. 1990. DNA restriction fragment length polymorphisms among *Mycosphaerella graminicola* (Anamorph *Septoria tritici*) isolates collected from a single wheat field. Phytopathology 80: 1368-373.

McMillan, W. O. and Bermingham, E. 1996. The phylogeographic pattern of mitochondrial DNA variation in the Dall's porpoise *Phocoenoides dalli*. Molecular Ecology 5:47-61.

Mes, J. J., Weststeijn, E. A., Herlaar, F., Lambalk, J. J. M., Wijbrandi, J., Haring, M. A. and Cornelissen, B. J. C. 1999. Biological and molecular characterization of *Fussarium oxysporum* f. sp. *lycopersici* divides race 1 isolates into separate virulence groups. Phytopathology 89: 156-160.

Michelmore, R. W. and Hulbert, S. H. 1987. Molecular markers for genetic analysis of phytopathogenic fungi. Annual Review of Phytopathology. 25: 383-404.

Michener, C. D. and Sokal, R. R. 1957. A quantitative approach to a problem in classification. Evolution. 11: 130-162.

Mills, S. D., Förster, H. and Coffey, M. D. 1991. Taxonomic structure of *Phyophthora cryptogea* and *P. drechsleri* based on isozyme and mitochondrial DNA analyses. Mycological Research 95: 31-48.

Nadler, S. A., Lindquist, R, L. and Near, T. J. 1995. Genetic structure of midwestern *Ascaris suum* population: a comparison of isoenzyme and RAPD markers. Journal of Parasitology 81:385-394.

Newton, A. C. 1987. Markers in pathogen populations. In: Day, P. R. and Jellis, G. J. (eds.), Genetics and plant pathogenesis, Blackwell Scientific, Oxford, UK.

Orolaza, N. P., Lamari, L., and Ballance, G. M. 1995. Evidence of a host-specific chlorosis toxin from *Pyrenophora tritici-repentis*, the causal agent of tan spot of wheat. Phytopathology 85: 1282-1287.

O'Neill, N. R., Van-Berkum, P., Lin, J. J., Kuo, J., Ude, G. N., Kenworthy, W. and Saunders, J. A. 1997. Application of amplified restriction fragment length polymorphism for genetic characterization of *Colletotrichum* pathogens of alfalfa. Phytopathology. 87: 745-750.

Powell, W., Morgante, M., Andre, C., Hanafey, M., Vogel, J., Tingey, S. and Rafalski, A. 1996. Comparison of RFLP, RAPD, AFLP and SSR (microsatellite) markers for germplasm analysis. Molecular Breeding 2:225-238.

Provan, J., Powell, W. and Waugh, R. 1996. Microsatellite analysis of relationship within cultivated potato (*Solanum tuberosum*). Theoretical and Applied Genetics 92: 1078-1084.

Rees, R. G. and Platz, G. J. 1992. Tan spot and its control-some Australian experiences. Pages 1-9 in: Advances in Tan Spot Research- Proceedings of the Second International Tan Spot Workshop. L. J. Francl, J. M. Krupinsky, M. P. McMullen eds. North Dakota State University.

Rees, R. G., Platz, G. J. and Mayer, R. J. 1982. Yield losses in wheat from yellow spot: comparison of estimates derived from single tillers and spots. Australian Journal of Agricultural Research 33: 899-908.

Rees, R. G., and Platz, G. J. 1980. The epidemiology of yellow spot in southern Queensland. Australian Journal of Agricultural Research 31:259-267.

Saitou, N. and Nei, M. 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Molecular Biology and Evolution 4: 406-425.

Schilder, A. M. C., and Bergstrom, G. C. 1990. Variation in virulence within the population of *Pyrenophora tritici-repentis* in New York. Phytopathology 80: 84-90.

Schilder, A. M. C., and Bergstrom, G. C. 1992. The dispersal of conidia and ascospores of *Pyrenophora tritici-repentis*. In: L. J. Francl, J. M. Krupinsky, and M. P. McMullen, eds., Advances in Tan Spot Research. North Dakota Agric. Exp. Station, Fargo.

Shi, Y. L., Loomis, P., Christian, D., Carris, L. M. and Leung, H. 1996. Analysis of the genetic relationships among the wheat bunt fungi using RAPD and ribosomal DNA markers. Phytopathology 86:311-318.

Shoemaker, R. A. 1962. Drechslera Ito. Canadian Journal of Botany 40: 809-836.

Stammler G., Seemüler, E. and Duncan, J. M. (1993). Analysis of RFLPs in nuclear and mitochondrial DNA and the taxonomy of *Phytophthora fragariae*. Mycological Research. 97: 150-156.

Strelkov, S. E., Lamari, L., and Ballance, G. M. 1999. Characterization of a host-specific protein toxin (Ptr ToxB) from *Pyrenophora tritici-repentis*. Molecular Plant-Microbe Interactions 12: 728-732.

Swofford, D. L. and Olsen, G. J. 1990. Phylogeny reconstruction. Pages 411-501 in: Molecular Systematics. D. M. Hillis, and C. Moritz (eds.) Sinauer Associates, Inc. Publishers, Sunderland, Massachusetts, USA.

Takauz, A. 1976. Distribution, severity, and relative importance of leaf spot diseases of wheat in western Canada in 1974. Canadian Plant Disease Survey 56:36-40.
Fargo. 142 pp.

Tingey, S. V. and del Tufo, J. P. 1993. Genetic Analysis with Random Amplified Polymorphic DNA markers. Plant Physiology 101: 349-352.

Tomás, A., and Bockus, W. W. 1987. Cultivar specific toxicity of culture filtrate of *Pyrenophora tritici-repentis*. Phytopathology 77: 1337-1366.

Tomás, A., Feng, G. H., Bockus, W. W. and Leach, J. E. 1990. Purification of a cultivar-specific toxin from *Pyrenophora tritici-repentis*, causal agent of tan spot of wheat. Molecular Plant-Microbe Interactions 3: 221-224.

Tuori, R. P., Wolpert, T. J. and Ciuffetti, L. M. 1995. Purification and immunological characterization of toxin components from cultures of *Pyrenophora tritici-repentis*. Molecular Plant-Microbe Interactions 8: 41-48.

Vakalounakis, D. J. and Fragkiadakis, G. A. 1999. Genetic diversity of *Fusarium oxysporum* isolates from cucumber: Differentiation by pathogenicity, vegetative compatibility, and RAPD fingerprinting. Phytopathology 89: 161-168.

Vailejos, C. E. 1993. Enzyme activity staining. In: Tanksley, SD. and Orton, T. J. (eds.), Isozymes in plant genetics and breeding, Part A, Elsevier Science Publishers B. V., Amsterdam, Netherlands.

Van der Lee, T., De witte, T., Drenth, A., Alfonso, C. and Govers, F. (1997). AFLP linkage map of the oomycete *Phytophthora infestans*. Fungal Genetics and Biology 21: 278-291.

Vos, P., Hoger, R., Bleeker, M., Reijans, M., Van de Lee, T., Homes, M., Frijters, A., Pot, J., Peleman, J., Kuiper, M. and Zabeau, M. (1995). AFLP: a new technique for DNA fingerprinting. Nucleic Acids Research 23: 4407-4414.

Wehmeyer, L. E. 1954. Perithecial development in *Pleospora trichostoma*. Botanical Gazette 115:297-310.

Weiland, J. J., Steffenson, B. J., Cartwright, R. D. and Webster, R. K. 1999. Identification of molecular genetic markers in *Pyrenophora teres* f. *teres* associated with low virulence on 'Harbin' barley. Phytopathology 89: 176-181.

Welsch, J. and McClelland, M. 1990. Fingerprinting genomes using PCR with arbitrary primers. Nucleic Acids Research 18: 7213-7218.

Welz, H. G., Köhler, W., and Leonard, K. J. 1993. Isozyme variation within and among pathogenic races of *Cochiobolus carbonum* on Corn in North Carolina. Phytopathology 84: 31-38.

White, G. and Powell, W. 1997. Isolation and characterization of microsatellite loci in *Swietenia humilis* (Meliaceae): an endangered tropical hardwood species. Molecular Ecology 6: 851-860.

White, J. L. and Kaper, J. M. 1989. A simple method for detection of viral satellite RNAs in small plant tissue samples. Journal of Virological Methods 23: 83-94.

Wiese, M. V. 1987. Compendium of wheat diseases. Pages 42-46 American Phytopathological Society Press, St. Paul, Minnessota.

Williams, J. G. K., Kubelik, A. R., Livak, K. J., Rafalski, J. A. and Tingey, S. V. 1990. DNA polymorphisms amplified by arbitrary primers are useful as genetic markers. Nucleic Acids Research 18: 6531-6535.

Wright, K. H., and Sutton, J. C. 1990. Inoculum of *Pyrenophora tritici-repentis* in relation to epidemics of tan spot of wheat in Ontario. Canadian Journal of Plant Pathology 12:149-175.

Wang, X-R., Ennos, R. A., Szmidt, A. E. and Hansson, P. 1997. Genetic variability in the canker pathogen fungus, *Gremmeniella abietina*. 2. Fine-scale investigation of the population genetic structure. Canadian Journal of Botany 75:1460-1469.

Zambino, P. J., and Szabo, L. J. 1993. Phylogenetic relationships of selected cereal and grass rusts based on rDNA sequence analysis. Mycologia 85: 401-414.

Zhang, H., Francl, L. J., Jordahl, J. G. and Meinhardt, S. W. 1997. Structural and physical properties of a necrosis-inducing toxin from *Pyrenophora tritici-repentis*. Phytopathology 87: 154-160.

6. APPENDICES

Appendix 1. Culture media
Appendix 1a. V-8 PDA medium

PDA	10.0 gm
$CaCO_3$	3.0 gm
Agar	10.0 gm
V8 juice	150.0 ml
Distilled H_2O	850.0 ml

Reference:
Lamari, L. and Bernier, C. C. 1989. Evaluation of wheat for reaction to tan spot (*Pyrenophora tritici-repentis*) based on lesion type. Can. J. Plant Pathol. 11:49-56.

Appendix 1b. Fries medium

NH_4 Tartrate	5.0 gm
NH_4NO_3	1.0 gm
$MgSO_4.7H_2O$	0.5 gm
KH_2PO_4	1.3 gm
Sucrose	30.0 gm
Yeast Extract	1.0 gm
Trace Element	2.0 ml
Distilled H_2O	850 ml

Trace Element

LiCl	167 mg
$CuCl_2$	107 mg
H_2MoO_4	34 mg
$MnCl_2.4H_2O$	72 mg
$CoCl_2.4H_2O$	80 mg

Reference:
Dhingra, O. D., and Sinclair, J. B. 1985. Basic Plant Pathology Methods. CRC Press, Boca Raton, Florida. pp.355.

Appendix 2. DNA extraction buffer (200 ml)

0.1 M Glycine	1.5 gm
50 mM NaCl	2 ml of 5M NaCl
10 mM EDTA	2 ml of 0.5M EDTA
2% SDS	20 ml of 20% SDS
1% Sodium Lauryl sarcosine	2 gm
Distilled H_2O	176 ml
Adjust PH to 9.0 with NaOH	

Reference:

White, J. L. and Kaper, J. M. 1989. A simple method for detection of viral satellite RNAs in small plant tissue samples. Journal of Virological Methods 23: 83-94.

Appendix 3. AMOVA output (nested analysis)

```
Wednesday 11/15/2000                                              11h46'

                        ANALYSIS OF MOLECULAR VARIANCE
                        ==============================

Input files
===========

 Distance Matrix File           : D:\PROGRA~1\AMOVA155\NESTED.DIS
 Group File                     : D:\PROGRA~1\AMOVA155\NESTED.GRP
 Population File generic name   : nested

 No. of Groups       :     2
 No. of Populations  :     5
 No. of Chromosomes  :    51
 No. of Haplotypes   :    49

Sum of squares
==============
                    Nested                      Among Population
Among Group

                  SS(A) =      53.1322     SSAmPop =    80.1152     SSAmGrp
=    53.1322
                  SS(B) =      26.9830     SSWiPop =   158.3554     SSWiGrp
=   185.3384
                  SS(C) =     158.3554
                  ====================
SS(A) + SS(B) + SS(C) =     238.4706
            SS(Total) =     238.4706

Within Population sums of squares
=================================
 (Population    1)       53.6471
 (Population    2)       45.8750
 (Population    3)       41.5000
 (Population    4)       13.3333
 (Population    5)        4.0000


Within Group sums of squares
============================

 (Group    1)      103.3939
 (Group    2)       81.9444


Mean squares
============

                                  Nested              Among Population
         Among Group
 d.f.(A) =       1     MS(A) =    53.132     MS(A)(    4 df) =    20.029
MS(A)(   1 df) =   53.132
 d.f.(B) =       3     MS(B) =     8.994     MS(B)(   46 df) =     3.443
MS(B)(  49 df) =    3.782
```

```
 d.f.(C) =     46     MS(C) =     3.443
          =====
 Total         50

Variance components
===================
                          Nested Analysis
 Variance among groups                          V(A) :         1.84235559990
(31.18%)
 Variance among populations within groups       V(B) :         0.62363507734
(10.55%)
 Variance within populations                    V(C) :         3.44250852510
(58.26%)

 PHI-statistics :    PHIst =  0.417    PHIsc =  0.153    PHIct =  0.312

                    Analysis Among Populations
 Variance among populations                     V(A) :         1.76597247440
(33.91%)
 Variance within populations                    V(B) :         3.44250852510
(66.09%)

                    Analysis Among Groups
 Variance among groups                          V(A) :         2.11855152220
(35.90%)
 Variance within groups                         V(B) :         3.78241599670
(64.10%)


Distances among populations
===========================

 Distances = PhiST between pairs of populations

 Above diagonal      : Probability  Random distance (PhiST) > Observed
distance
 Nb. of iterations :   1000

       1       2       3       4       5
  0.0000  0.3886  0.0000  0.0000  0.0000
  0.0123  0.0000  0.0000  0.0000  0.0000
  0.3993  0.4575  0.0000  0.0000  0.0000
  0.3991  0.4590  0.3139  0.0000  0.0000
  0.4040  0.4632  0.1961  0.1896  0.0000


 Distances = Modified Coancestry coefficient (-ln[1-PhiST]=t/2N) between
pairs of populations

 Above diagonal      : Probability  Random distance (PhiST) > Observed
distance
 Nb. of iterations :   1000

       1       2       3       4       5
  0.0000  0.3886  0.0000  0.0000  0.0000
  0.0124  0.0000  0.0000  0.0000  0.0000
  0.5097  0.6116  0.0000  0.0000  0.0000
  0.5093  0.6144  0.3767  0.0000  0.0000
  0.5174  0.6221  0.2183  0.2102  0.0000
```

Variance Components Significance
==========

No. of permutations : 1000

	P(More extreme random value)	No. of different values in distribution
VarA and PhiCT :	<0.0010	10
VarB and PhiSC :	<0.0010	71
VarC and PhiST :	<0.0010	64

Appendix 4. AMOVA output (among groups analysis)

```
Tuesday  1/ 9/2001                                                  15h 9'

                        ANALYSIS OF MOLECULAR VARIANCE
                        ==============================

Input files
===========

 Distance Matrix File          : D:\PROGRA~1\AMOVA155\BETGP.DIS
 Group File                    : D:\PROGRA~1\AMOVA155\BETGP.GRP
 Population File generic name  : bet-gp

 No. of Groups       :     1
 No. of Populations  :     2
 No. of Chromosomes  :    51
 No. of Haplotypes   :    49

Sum of squares
==============

               SS(A) =      53.1322
               SS(B) =     185.3384
               ====================
          SS(Total) =     238.4706

Within Population sums of squares
=================================
 (Population    1)      103.3939
 (Population    2)       81.9444


Within Group sums of squares
============================

 (Group    1)      238.4706


Mean squares
============

 d.f.(A) =      1    MS(A) =   53.132
 d.f.(B) =     49    MS(B) =    3.782
           =====
 Total         50

Variance components
===================

 Variance among populations                V(A) :        2.11855152220
(35.90%)
 Variance within populations               V(B) :        3.78241599670
(64.10%)

 PHI-statistics :   PHIst =  0.359


Distances among populations
===========================
```

Distances = PhiST between pairs of populations

Above diagonal : Probability Random distance (PhiST) > Observed distance
Nb. of iterations : 1000

1	2
0.0000	0.0000
0.3590	0.0000

Distances = Modified Coancestry coefficient (-ln[1-PhiST]=t/2N) between pairs of populations

Above diagonal : Probability Random distance (PhiST) > Observed distance
Nb. of iterations : 1000

1	2
0.0000	0.0000
0.4448	0.0000

Variance Components Significance
==========

No. of permutations : 1000

	P(More extreme random value)	No. of different values in distribution
VarA and PhiST :	<0.0010	93

	primer 601			primer 602									
	2kb	780bp	600bp	2kb	2.1kb	1.8kb	1.65kb	1.55kb	1.27kb	1.15kb	1.08kb	967bp	889bp
SC24-3	0	0	0	1	0	0	0	1	0	0	0	1	0
SC29-2	0	0	0	1	0	0	0	1	0	0	0	1	1
SC10-1	0	0	0	1	0	0	0	1	0	1	0	1	0
92-171	0	0	0	1	1	0	1	1	0	0	0	1	0
ASCI	0	0	0	1	0	0	0	1	0	0	0	1	0
98MD5	0	0	0	1	0	0	0	1	0	0	0	1	0
98MD7	0	0	0	1	0	0	0	1	0	0	0	1	0
98MD8	0	0	0	1	0	0	0	1	0	0	0	1	0
98MD10	0	0	0	1	0	0	0	1	0	0	0	1	0
98MD11	0	0	0	1	0	0	0	1	0	1	0	1	0
94-12	0	0	0	1	0	0	0	1	0	0	0	1	0
11JA	0	0	0	1	0	0	0	1	0	0	0	1	0
15JA	0	1	1	1	0	0	0	1	0	0	0	1	0
4JS	0	0	0	1	0	0	0	1	0	0	0	1	1
Jeff-pt98	0	0	0	1	0	0	0	1	0	1	0	1	0
PDY7	0	0	0	1	0	0	0	1	0	0	0	1	0
FH86	1	0	0	1	0	0	0	1	0	0	0	1	0
SC3-11	0	0	0	1	0	0	0	1	0	0	0	1	1
SC3-4	0	0	0	1	0	0	0	1	0	0	0	1	1
SC24-2	0	0	0	1	0	0	0	1	0	0	0	1	0
331-3	0	0	0	0	0	1	0	0	1	0	0	1	0
91-18	0	0	0	1	0	0	0	1	0	0	0	1	0
44F	0	0	0	1	0	0	0	1	0	0	0	1	0
32JA	0	0	0	1	0	0	0	1	0	0	0	1	0
92-40J	0	0	0	1	0	0	0	1	0	0	0	1	0
91-52	0	0	0	1	0	0	0	1	0	0	0	1	0
90-27	0	0	0	1	0	0	0	1	0	0	0	0	1
92-28J	0	0	0	1	0	0	0	1	0	1	0	0	1
94-106	0	0	0	1	0	0	0	1	0	0	0	1	0

Appendix 5

	primer 601			primer 602									
	2kb	780bp	600bp	2kb	2.1kb	1.8kb	1.65kb	1.55kb	1.27kb	1.15kb	1.08kb	967bp	889bp
90-68	0	0	0	1	0	0	0	1	0	1	0	1	1
91-46	0	0	0	1	0	0	0	1	0	0	0	1	0
92-43	0	0	0	1	0	0	0	1	0	0	0	1	0
94-9M	0	0	0	1	0	0	0	1	0	0	0	0	1
86-124	0	0	0	1	0	0	0	1	0	0	0	1	0
331-9	0	0	0	0	0	1	0	0	0	0	0	1	1
331-2	0	0	0	1	0	0	0	1	0	0	0	1	0
94-25	0	0	0	1	0	0	0	1	0	0	0	1	0
94-115	0	0	0	1	0	0	0	0	1	0	0	1	0
D308	0	0	0	1	0	0	0	0	1	0	0	1	0
SC22-2	0	0	0	1	0	0	1	0	0	0	0	1	0
SC29-1	1	0	0	1	0	0	1	0	0	0	0	1	0
SC29-9	1	0	0	1	0	0	1	0	0	0	0	1	0
SC29-8	1	0	0	1	0	0	1	0	0	0	0	1	0
90-2	0	0	1	1	1	0	1	1	0	0	0	1	0
49JA	0	0	0	1	0	0	1	1	0	0	0	1	0
Alg3-24	0	0	0	1	0	0	0	0	0	0	0	1	0
Alg3-X3	0	0	0	1	0	0	0	0	0	0	0	1	0
Alg3-X1	0	0	0	1	0	0	0	0	0	0	0	1	0
Alg4-X1	0	0	0	0	0	0	1	0	0	0	0	1	0
Alg7-X3	0	0	0	1	0	0	0	0	0	0	0	1	0
Alg5-X1	0	0	0	1	0	0	0	0	0	0	1	1	0
AlgH1	0	0	0	1	0	0	0	0	0	0	0	1	0
AlgH2	0	0	0	1	0	0	0	0	0	0	0	1	0

Appendix 5

	primer 609							primer 649							
	2kb	1.7kb	1.6kb	1.3kb	1.25kb	1.2kb	1kb	2kb	1.5kb	1.4kb	1.15kb	1.1kb	800bp	700bp	500bp
SC24-3	0	1	1	1	0	1	0	1	1	1	0	0	1	0	0
SC29-2	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
SC10-1	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
92-171	0	0	1	1	0	0	0	0	1	1	0	0	1	0	0
ASCI	0	1	1	1	0	1	0	1	1	1	0	0	1	0	0
98MD5	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
98MD7	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
98MD8	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
98MD10	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
98MD11	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
94-12	0	1	1	1	0	1	0	1	1	1	0	0	1	0	0
11JA	0	1	1	1	0	1	0	0	1	1	1	0	1	0	0
15JA	0	1	1	1	0	1	0	1	0	1	0	0	0	0	0
4JS	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
Jeff-pt98	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
PDY7	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
FH86	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
SC3-11	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
SC3-4	0	1	1	1	1	0	0	1	1	1	0	0	1	0	0
SC24-2	0	1	1	1	0	1	0	1	1	1	0	0	1	0	0
331-3	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0
91-18	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
44F	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
32JA	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
92-40J	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
91-52	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
90-27	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
92-28J	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
94-106	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0

Appendix 5

	primer 609							primer 649							
	2kb	1.7kb	1.6kb	1.3kb	1.25kb	1.2kb	1kb	2kb	1.5kb	1.4kb	1.15kb	1.1kb	800bp	700bp	500bp
90-68	0	1	1	1	0	0	0	0	1	1	0	0	1	0	0
91-46	0	0	0	0	0	0	1	0	1	1	0	0	1	0	0
92-43	0	1	1	1	0	1	0	0	1	1	0	0	1	0	0
94-9M	0	1	1	1	1	0	0	0	1	1	0	0	1	0	0
86-124	0	1	1	1	1	0	0	0	1	1	0	0	1	0	0
331-9	0	1	1	1	0	0	0	0	1	1	0	0	0	0	0
331-2	0	1	1	1	0	0	0	0	1	1	0	0	0	0	0
94-25	0	1	1	1	0	0	0	0	1	1	0	0	0	0	0
94-115	0	1	1	1	0	1	0	0	1	1	0	0	0	0	0
D308	1	0	1	1	0	0	0	0	1	1	0	0	0	0	0
SC22-2	1	1	1	1	0	1	0	1	1	1	0	0	0	1	0
SC29-1	1	1	1	1	0	0	0	0	1	1	0	0	0	1	0
SC29-9	1	1	1	1	0	0	0	0	1	1	0	0	0	1	0
SC29-8	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0
90-2	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0
49JA	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0
Alg3-24	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0
Alg3-X3	0	1	1	1	0	0	0	1	1	1	0	1	0	1	0
Alg3-X1	0	1	1	1	0	1	0	1	1	1	0	1	0	1	0
Alg4-X1	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0
Alg7-X3	0	1	1	1	0	0	0	0	1	1	0	0	0	0	0
Alg5-X1	0	1	1	1	1	0	0	0	1	1	0	1	0	0	0
AlgH1	1	1	1	1	0	0	0	0	1	1	0	0	0	0	0
AlgH2	0	1	1	1	0	0	0	0	1	1	0	1	0	0	0

Appendix 5

					primer 650					primer 638		
	1.5kb	1.6kb	1.25kb	1.7 kb	875bp	800bp	750bp	650bp	500bp	800bp	750bp	700bp
SC24-3	0	0	0	0	0	0	0	0	1	0	0	1
SC29-2	0	0	0	0	0	0	0	0	1	0	0	1
SC10-1	0	0	0	0	0	0	0	0	1	0	0	1
92-171	0	1	0	0	0	0	0	0	1	0	0	1
ASCI	0	1	0	0	0	0	0	0	1	0	0	1
98MD5	0	0	0	0	0	0	0	1	1	0	0	1
98MD7	0	0	0	0	0	0	0	0	1	0	0	1
98MD8	0	0	0	0	0	0	0	1	1	0	0	1
98MD10	0	0	0	0	0	0	0	1	1	0	0	1
98MD11	0	0	0	0	0	0	0	0	1	0	0	1
94-12	0	1	0	0	0	0	0	0	1	0	0	1
11JA	0	1	0	0	0	0	0	0	1	0	0	1
15JA	0	1	1	0	0	0	0	0	1	0	0	1
4JS	0	0	0	0	0	0	0	0	1	0	0	1
Jeff-pt98	0	0	0	0	0	0	0	0	1	0	0	1
PDY7	0	0	0	0	0	0	0	0	1	0	0	1
FH86	0	0	0	0	0	0	0	0	1	0	0	1
SC3-11	0	1	0	0	0	0	0	0	1	0	0	1
SC3-4	0	1	0	0	0	0	0	0	1	0	0	1
SC24-2	0	0	0	0	0	0	0	0	1	0	0	1
331-3	0	0	0	0	0	0	0	0	1	0	0	1
91-18	0	1	0	0	0	0	0	0	1	0	0	1
44F	0	0	0	0	0	0	0	0	1	0	0	1
32JA	0	0	0	0	0	1	0	0	1	0	0	1
92-40J	0	1	0	0	0	0	0	0	1	0	0	1
91-52	0	0	0	0	0	0	0	0	1	0	0	1
90-27	0	1	0	0	0	0	0	0	1	0	0	1
92-28J	0	0	0	0	0	0	0	0	1	0	0	1
94-106	0	0	0	0	0	0	0	0	1	0	0	1

Appendix 5

					primer 650					primer 638		
	1.5kb	1.6kb	1.25kb	1.7 kb	875bp	800bp	750bp	650bp	500bp	800bp	750bp	700bp
90-68	0	0	0	0	0	0	0	1	1	0	0	1
91-46	0	1	0	0	0	0	0	0	1	0	0	1
92-43	0	1	0	0	0	0	0	0	1	0	0	1
94-9M	0	0	0	0	0	0	0	0	1	0	0	1
86-124	0	0	0	0	0	0	0	0	1	0	0	1
331-9	0	0	0	0	0	0	0	0	1	0	1	1
331-2	0	0	0	0	0	0	0	0	1	0	1	1
94-25	0	0	0	0	0	0	0	1	1	0	1	1
94-115	0	1	0	0	0	0	0	1	1	0	1	1
D308	0	0	0	0	0	0	0	0	1	0	0	1
SC22-2	1	0	0	0	0	0	0	0	1	0	1	1
SC29-1	1	1	0	0	0	0	0	0	1	1	0	1
SC29-9	1	0	0	0	0	0	0	0	1	1	0	1
SC29-8	1	0	0	0	0	0	0	0	1	1	0	1
90-2	0	1	1	0	0	0	1	0	0	0	1	0
49JA	1	1	0	0	0	0	1	0	0	1	0	0
Alg3-24	0	0	0	0	0	0	0	0	1	0	0	1
Alg3-X3	0	0	0	0	0	0	0	0	1	0	0	1
Alg3-X1	0	1	0	0	0	0	0	0	1	0	0	1
Alg4-X1	0	0	0	0	0	0	0	0	1	0	0	1
Alg7-X3	0	1	0	0	0	0	0	0	1	0	0	1
Alg5-X1	0	0	0	0	0	0	0	0	1	0	0	1
AlgH1	0	1	0	0	0	0	0	0	1	0	1	1
AlgH2	0	1	0	1	1	0	0	0	1	0	0	1

Appendix 5

	primer 648					primer 652					
	2.2kb	560bp	2.3kb	2kb	1.4kb	1.3kb	1.137kb	762bp	700bp	656bp	603bp
SC24-3	1	0	1	1	1	1	0	1	0	0	0
SC29-2	0	0	1	1	1	1	0	1	0	0	0
SC10-1	1	0	1	1	1	1	0	0	1	0	0
92-171	0	0	1	1	1	0	1	1	0	0	0
ASCI	0	0	1	1	1	0	0	1	0	0	0
98MD5	0	0	0	0	0	1	0	0	0	0	0
98MD7	0	0	1	1	0	1	0	0	0	0	0
98MD8	0	0	1	1	0	1	0	0	0	0	0
98MD10	0	0	1	1	0	1	0	0	0	0	0
98MD11	0	0	1	1	0	1	0	0	1	0	0
94-12	0	0	1	1	1	0	1	1	1	0	0
11JA	0	0	1	1	1	0	0	1	1	0	0
15JA	0	0	0	1	1	0	1	1	0	0	0
4JS	0	0	1	1	1	0	0	0	0	0	0
Jeff-pt98	0	0	1	1	1	0	0	1	0	0	0
PDY7	0	0	1	1	1	0	1	1	0	0	0
FH86	0	0	1	1	1	0	0	1	0	0	0
SC3-11	0	0	1	1	0	1	0	1	0	0	0
SC3-4	1	0	1	1	0	0	0	1	0	0	0
SC24-2	0	0	1	1	1	0	0	1	0	0	0
331-3	0	0	0	1	0	1	0	0	1	0	0
91-18	0	0	1	1	1	0	0	1	0	0	0
44F	1	0	1	1	1	0	1	1	1	0	0
32JA	0	1	1	1	1	0	0	0	0	0	0
92-40J	0	0	0	0	1	1	0	0	0	0	0
91-52	0	0	0	0	1	0	1	1	0	0	0
90-27	0	0	1	0	1	0	0	1	0	0	0
92-28J	0	0	1	1	0	1	0	0	0	0	0
94-106	0	0	1	0	0	0	0	0	0	0	0

Appendix 5

	primer 648					Primer 652					
	2.2kb	560bp	2.3kb	2kb	1.4kb	1.3kb	1.137kb	762bp	700bp	656bp	603bp
90-68	0	0	0	1	1	0	0	1	0	0	0
91-46	0	0	1	1	1	0	0	1	0	0	0
92-43	0	0	1	1	1	0	0	1	0	0	0
94-9M	0	0	1	1	1	0	0	1	0	0	0
86-124	0	0	1	1	1	0	0	1	0	0	0
331-9	0	0	0	1	0	0	0	0	1	0	0
331-2	0	0	0	1	0	0	0	0	1	0	0
94-25	0	0	0	1	0	0	0	0	1	0	0
94-115	0	0	0	1	1	0	0	1	0	0	0
D308	0	0	1	1	0	0	0	0	1	0	0
SC22-2	0	0	0	1	0	0	1	0	1	0	0
SC29-1	0	0	0	1	0	0	1	0	1	0	0
SC29-9	1	0	0	1	0	0	1	0	1	0	0
SC29-8	0	0	0	1	0	0	1	0	1	0	0
90-2	0	0	0	1	0	0	0	0	1	1	1
49JA	0	0	0	1	0	1	0	0	1	1	1
Alg3-24	0	0	1	1	0	0	1	0	0	0	0
Alg3-X3	0	0	1	1	0	0	1	0	1	0	0
Alg3-X1	0	0	1	1	0	0	1	0	1	0	0
Alg4-X1	0	0	1	1	0	0	1	1	0	0	0
Alg7-X3	0	0	1	1	0	0	1	0	1	0	0
Alg5-X1	0	0	1	1	0	0	1	1	0	0	0
AlgH1	0	0	1	1	0	0	0	0	1	0	0
AlgH2	0	0	1	1	0	0	1	1	1	0	0

Appendix 5

	452bp	primer 1.6kb	652 1.663kb	1.477kb	742bp	1.016kb	11.9kb	1.22kb
SC24-3	0	0	0	0	0	0	0	1
SC29-2	0	0	0	0	0	0	0	1
SC10-1	0	1	0	0	1	0	0	0
92-171	0	0	0	0	0	0	0	1
ASCI	0	0	0	0	0	0	0	1
98MD5	0	0	0	0	0	0	0	1
98MD7	0	0	0	0	0	0	0	1
98MD8	0	0	0	0	0	0	0	1
98MD10	0	0	0	0	0	0	0	1
98MD11	0	0	0	0	0	0	0	1
94-12	0	0	0	0	0	0	0	1
11JA	0	0	0	0	0	1	0	1
15JA	0	0	0	0	0	0	0	0
4JS	0	0	0	0	0	0	0	1
Jeff-pt98	0	0	0	0	0	0	0	1
PDY7	0	0	0	0	0	0	0	1
FH86	0	0	0	0	0	0	0	1
SC3-11	0	0	0	0	0	0	0	1
SC3-4	0	0	0	0	0	0	0	1
SC24-2	0	0	0	0	0	0	0	1
331-3	0	0	0	0	0	0	1	0
91-18	0	0	0	0	0	0	0	1
44F	0	0	0	0	0	0	0	1
32JA	0	0	0	0	0	0	0	1
92-40J	0	0	0	0	0	0	0	1
91-52	0	0	0	0	0	0	0	1
90-27	0	0	0	0	0	0	0	1
92-28J	0	0	0	0	0	0	0	1
94-106	0	0	0	0	0	0	0	1

Appendix 5

	Primer 452bp	652 1.6kb	1.663kb	1.477kb	742bp	1.016kb	11.9kb	1.22kb
90-68	0	0	0	0	0	0	0	1
91-46	0	1	0	0	0	0	0	1
92-43	0	0	0	0	0	0	0	1
94-9M	0	0	0	0	0	0	0	1
86-124	0	0	0	0	0	0	0	1
331-9	0	0	0	0	0	0	0	0
331-2	0	0	0	0	0	0	0	0
94-25	0	0	0	0	0	0	0	0
94-115	0	0	0	0	0	0	0	0
D308	0	0	0	0	0	0	0	0
SC22-2	0	0	0	0	1	0	0	0
SC29-1	0	0	0	0	1	0	0	0
SC29-9	0	0	0	0	1	0	0	0
SC29-8	0	0	0	0	1	0	0	0
90-2	0	0	1	1	1	0	0	1
49JA	1	0	1	0	1	0	0	0
Alg3-24	0	0	0	0	0	0	0	1
Alg3-X3	0	0	0	0	1	0	0	1
Alg3-X1	0	0	0	0	1	0	0	1
Alg4-X1	0	0	0	0	0	0	0	1
Alg7-X3	0	0	0	0	1	0	0	1
Alg5-X1	0	0	0	0	0	0	0	1
AlgH1	0	0	0	1	1	0	0	1
AlgH2	0	0	0	0	1	0	0	1

Appendix 5

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Synchronous Queuing: A Co-allocation Mechanism for Multimedia Enabled Grids

by

Farag Azzedin

A thesis

Submitted to the Faculty of Graduate Studies

in Partial Fulfillment of the Requirements

for the degree of

MASTER OF SCIENCE

Department of Computer Science
University of Manitoba
Winnipeg, Manitoba, Canada

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a MSc thesis entitled:

Synchronous Queuing: A Co-allocation Mechanism for Multimedia Enabled Grids

submitted by: *Mr. Farag Azzedin*

in partial fulfillment of the requirements for the degree of: *MSc*

Dr. M. Maheswaran, Advisor

Dr. A.N. Arnason

Dr. J. Rueda
TR Labs

Date of Oral Examination: *May 9, 2001*

The student has satisfactorily completed and passed the MSc Oral Examination.

Dr. M. Maheswaran, Advisor

Dr. Dereck Meek
Chair of MSc Oral

Dr. A.N. Arnason

Dr. J. Rueda
TR Labs

(The signature of the Chair does not necessarily signify that the Chair has read the complete thesis.)

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

SYNCHRONOUS QUEUING: A CO-ALLOCATION MECHANISM FOR MULTIMEDIA ENABLED GRIDS

BY

FARAG AZZEDIN

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

of

MASTER OF SCIENCE

Acknowledgement

In the name of Allah, the Merciful, the Compassionate. Praise be to Him for His creation and making me submissive to Him. Thanks to Him that He sent us the prophets to guide us to the straight path.

I express my gratefulness to my supervisor Prof. M. Maheswaran, whose amazing patience, infinite help in all aspects, and non-stop support made me achieve what I did not know I had in me. I can never thank him enough. I am also thankful to the thesis committee members, Prof. N. Arnason and Dr. J. Rueda, for being on my thesis committee. Also, I would like to thank Prof. N. Arnason for his useful suggestions and help during the simulation phase.

All the love to my mother (Fatima) and my father (Ahmed), who are always in my heart and mind, and whose my accomplishment in life is none but the result of Allah's answering to their prayers and supplication to Him for my success.

Finally, I express my heartfelt gratitude to my wife (Ameena), whom whatever words of thanks I say, they would not be enough to do justice to her. And to our children Muhammad, Ahmed, and Hala whom just remembering them makes me know what I want to do in life.

Abstract

Grid computing systems are being positioned as a computing infrastructure of the future that will enable the usage of wide-area network computing systems for a variety of challenging applications. The multimedia enabled Grid (MEG) is an extension of the Grid concept to support the deployment of multimedia services to meet the ever increasing demand for multimedia from users engaging in a wide range of activities such as scientific research, education, commerce, and entertainment. The MEG will provide several new services and sustain several enabling technologies to support multimedia.

To provide an adequate level of service to multimedia applications, it is often necessary to simultaneously allocate the resources including predetermined capacities from the interconnecting networks to the applications. The simultaneous allocation of resources is often referred to as co-allocation in the Grid literature.

In this thesis, I propose a novel scheme called synchronous queuing (SQ) for implementing co-allocation with quality of service (QoS) assurances in Grids. The SQ does not require advance reservation capabilities at the resources, which is a fundamental difference between SQ and the other existing schemes. I formally define the co-allocation problem and classify existing approaches based on a taxonomy that is presented here. Based on the taxonomy, I discuss the situations under which SQ can be used for co-allocation in MEGs. The SQ scheduler introduces new scheduling concepts such as the notion of accounting for the previous work, the notion of introducing

intraQueue and interQueue schedulers and the notion of calculating the co-allocation skew. Simulation studies performed to evaluate SQ indicate that it outperforms admission control-based scheme by a significant margin.

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

The deployment of faster networking infrastructures and the availability of powerful microprocessors have positioned network computing as a cost-effective alternative to the traditional computing approaches. The network computing systems can be grouped into various categories depending on the extent of the system and the performance of the interconnection media. For example, clusters of workstations are network computing systems that use commodity networks to create very tight and dedicated coupling among the nodes. Another example of network computing is the metacomputing initiatives on the Internet that attempt to harness the available resources to perform complex parallel applications such as prime number sieves. Motivated by the successes of such specialized efforts, researchers have started examining a more generalized resource/information sharing and integration infrastructure called the Grid [FoK99]. The Grid is defined as a generalized, large-scale computing and data handling virtual system that is formed by aggregating the services provided by several distributed resources [BaB00, FoK98, KrM00, JoG99, MaK00]. A Grid can potentially provide pervasive, dependable, consistent, and cost-effective access to the diverse services provided by the distributed resources and support problem solving environments that may be constructed using such resources.

The *multimedia enabled Grid* (MEG) is an extension of the Grid concept to support the deployment of multimedia services to meet the ever-increasing demand for multimedia from users engaging in a wide range of activities such as scientific research, education,

commerce, and entertainment. The MEG will provide various new services and sustain several enabling technologies to support multimedia. Some of the new services include transparent user profile location and access supporting "upcalls" so that adaptive applications can be developed. A user in such an environment is not tied to a specific machine but rather is a machine independent entity that exists in the Grid and can transparently carry its profile across the different platforms constituting the Grid. Some of the enabling technologies that will be supported include: (a) *quality of service* (QoS), (b) multicast, (c) streaming data, (d) co-allocation of resources, and (e) resource discovery.

Multimedia applications (e.g., digital audio or video) are known to impose real-time requirements on the underlying computing and communication systems [NiL97, AzM00]. Some of these applications require multiple networked resources for their execution. To provide an adequate level of service to the users, it is often necessary to allocate these resources including predetermined capacities from the interconnecting networks simultaneously to the particular applications. Examples of applications that require simultaneous allocation of resources include multimedia conferencing, virtual reality based distributed interactive simulation, distance learning, etc. The simultaneous allocation of resources is referred to as *co-allocation* in the Grid literature.

The co-allocation in a MEG environment is a much more general problem than that in traditional distributed multimedia systems. This is due to various issues including: (a)

location independent access and management of resources, (b) resource heterogeneity both in terms of capability and policy, and (c) geographically distributed location of the resources. These issues call for a resource management model with a hierarchical scheduling structure. The hierarchical scheduling structure introduces "hidden" scheduling [YaL96] problems rendering the overall resource management and particularly the co-allocation of resources a challenging task.

Most existing approaches [FeG97, FoK99] to co-allocation in wide-area distributed systems depend on the ability of the resources to support advance reservations. While performing co-allocation via advance reservation simplifies the problem, this approach has several drawbacks. One of the drawbacks is that this model does not allow the over subscription of the resources and thus leading to under utilization of the overall system. Another drawback is that the advance reservation-based approach imposes strict timing constraints on the client side.

This thesis presents *synchronous queuing* (SQ) a novel scheme for co-allocation that does not require advance reservation capabilities at the resources. The scheme provides co-allocation with QoS constraints, i.e., it is possible to perform co-allocation with hard QoS guarantees as well as co-allocation with best-effort guarantees.

The thesis is organized as follows: Section 2 presents the notation and mathematically defines the co-allocation problem. A taxonomy of existing approaches to perform co-

allocation in distributed systems is presented in Section 3. Section 4 examines the related work. Section 5 sketches in detail the *synchronous queuing* (SQ) scheme to solve the co-allocation problem in MEGs. This is followed by simulation results in Section 6. Section 7 summarizes the thesis and presents directions for future work.

2 NOTATION AND PROBLEM DEFINITION

One of the distinguishing features of the Grid concept is the recognition of the heterogeneity [ScN99] and site autonomy issues that are faced by the ultra-large scale distributed systems. One of the ways Grids handle these issues is to use a scheduling hierarchy. Figure 1 shows a basic block diagram for the resource management architecture for the Grid system with a two-level scheduling hierarchy consisting of local schedulers and Grid-level schedulers.

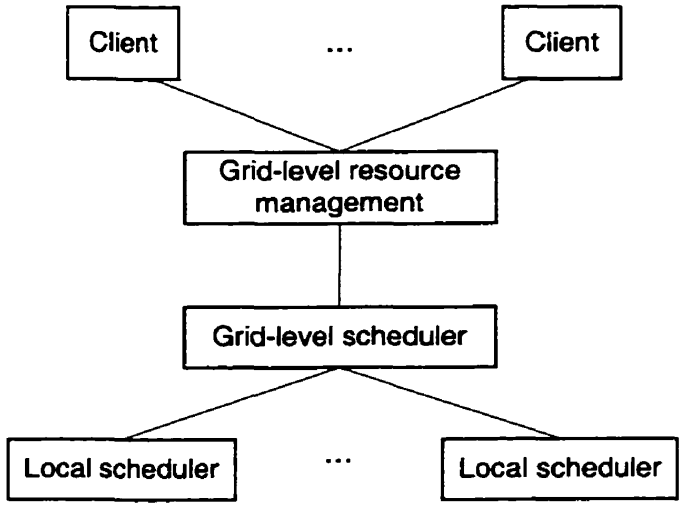


Figure 1: A block diagram for an overall resource management architecture for the Grids.

Let t denote a task submitted by a client to the Grid for processing and let this task t be composed of n subtasks $s_0,...,s_{n-1}$. Consider the situation where a Grid-level scheduler maps the different subtasks to different machines in the Grid. The Grid-level schedulers assign to a particular machine various tasks and subtasks, which are further scheduled by the local scheduler that controls the machine in a timeline fashion as illustrated in Figure

2. Some of these tasks and subtasks might have co-allocation requirements and others may not. t_0

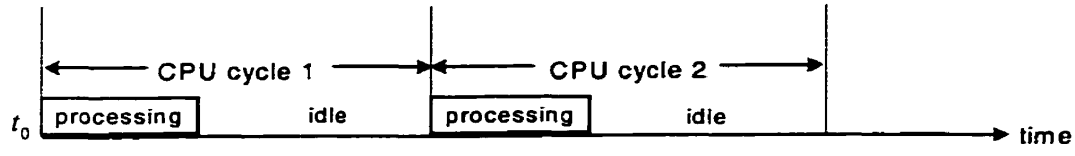


Figure 2: A local scheduler's periodic timing diagram.

Once the subtasks $s_0,...,s_{n-1}$ of task t are assigned to the different local schedulers, it the responsibility of the local schedulers to allocate sufficient machine resources (e.g., CPU cycles) to execute each subtask. Furthermore, let $a_0,a_1,...,a_{n-1}$ be the threads that are instantiated at the local machines for the subtasks $s_0,...,s_{n-1}$, respectively. Because the different local schedulers will have different mix of tasks and subtasks their behavior will be different. Note that because task t has co-allocation requirements all its subtasks *must*

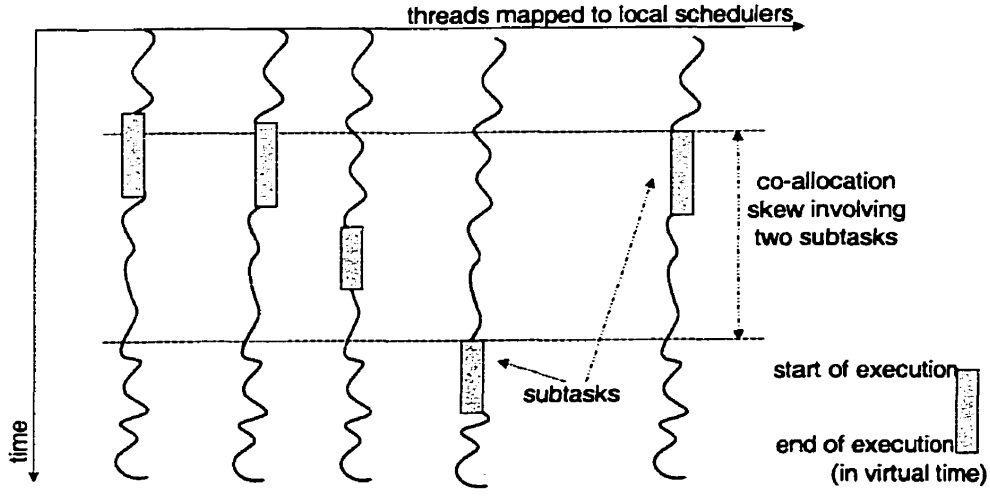


Figure 3: Example scenario that causes a co-allocation skew.

proceed with their execution simultaneously. Some of these subtasks might be delayed before they are allocated sufficient resources. This delay is referred to as *co-allocation skew*. The *co-allocation skew* involving two subtasks is illustrated in Figure 3. The goal of the synchronous queuing algorithm is to minimize this *co-allocation skew* for all applications that require co-allocation.

Consider two subtasks s_i and s_j. Assume that they become runnable at the 1^{st} schedule cycle. For the rest of the thesis, the term "CPU bandwidth" means the total CPU cycles per second available. Each thread a_i will be asking of a *share* of the local machine's CPU bandwidth. This *share* is expressed as a weight (explained in more detail in Section 5.5) assigned to thread a_i. Let r_{a_i} be the weight of thread a_i, m be the number of schedule cycles to date, and $W_k^{a_i}$ be the work done by thread a_i at the k^{th} schedule cycle. Then, threads a_i and a_j are said to be synchronized if, for any k^{th} schedule cycle the aggregate work done normalized by weight since the two threads a_i and a_j became runnable are identical (i.e., $\left| \frac{\sum_{k=1}^{m} W_k^{a_i}}{r_{a_i}} - \frac{\sum_{k=1}^{m} W_k^{a_j}}{r_{a_j}} \right| = 0$). Clearly, this is an idealized definition of synchronization that assumes infinitely divisible subtasks. Hence, the objective of synchronous queuing is to minimize the difference as close to zero as possible (i.e., $\left| \frac{\sum_{k=1}^{m} W_k^{a_i}}{r_{a_i}} - \frac{\sum_{k=1}^{m} W_k^{a_j}}{r_{a_j}} \right| \leq threshold$, for all $i, j,\ i \neq j$).

3 A TAXONOMY OF EXISTING APPROACHES

Traditionally, the Internet was used for running elastic applications for which it was sufficient to provide one single service class known as "best-effort" service. Elastic applications are those that can adjust, over wide ranges, to changes in delay and throughput across an internet and still meet their needs [Sta97]. In the MEG, there will be different types of applications from various application domains for which the "best-effort" service is inadequate. Because MEG is a specialization of the Grid concept [FoK99] to the multimedia applications, supporting various classes of quality of service is essential.

Classifying these wide ranges of applications that might co-exist in the MEG environment is a key element in determining a suitable algorithm to solve the co-allocation problem. In the MEG system, it is not uncommon for a user to run non-real and real time jobs simultaneously. Figure 4 shows a classification of the various applications that might co-exist in a MEG system.

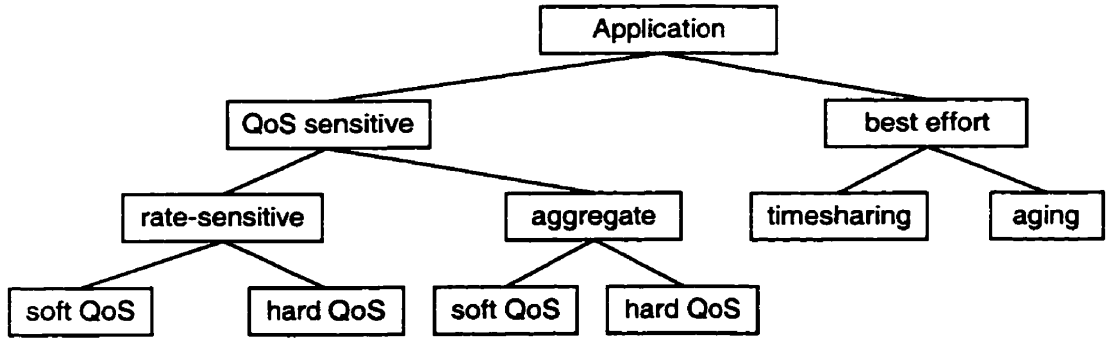


Figure 4: Different classes of applications.

- *rate-sensitive applications:* are applications that depend on accomplishing (finishing) a consistent x amount of work per t amount of time throughout the application's lifetime.
- *aggregate applications:* are applications that depend on accomplishing (finishing) x amount of work per t amount of time throughout the application's lifetime. The amount of work established per period of time t varies (i.e. is not consistent). The emphasis here is on finishing the task by its deadline.
- *hard QoS applications:* are application with stringent progress constraints [YaD99], for which missing a deadline for these applications leads to catastrophic failures. These applications require a deterministic guarantee for their QoS parameters.
- *soft QoS applications:* missing a deadline for these applications only diminishes the quality of the results and does not lead to catastrophic failures. These applications require a statistical or probabilistic guarantee for their QoS parameters.

Different approaches exist to solve the allocation and co-allocation of resources to this wide range of applications. An allocation/co-allocation approach can be summarized based on the following properties: (a) the specific types (i.e. best effort and/or QoS sensitive) of applications the scheme can accommodate, (b) the environment (i.e., single or distributed) in which the scheme can be implemented, (c) the scheme's technique can be specific to one or particular mixes of resources, and do not extend easily to other

resources. Or it can be generalized to manage many diverse resources, and (d) the allocation/co-allocation scheme can schedule all the tasks in the system based on proportional sharing, priorities, or support such a generality by introducing hierarchical schedulers as a tool to support a variety of QoS sensitive as well as best-effort applications.

4 RELATED WORK

4.1 Overview

In recent years, a lot of research has been done on finding ways for resource reservation schemes to accommodate the increasing demand for deploying real-time (multimedia) applications. The reservation models are mostly concentrated on *immediate reservation* where the reserved resources are scheduled immediately. However, providing QoS guarantees to *immediate reservation* schemes is a difficult task simply because tasks are competing for resources' availability. Furthermore, *immediate reservation* scheme's admission decision is made based on the resource availability at hand and hence under utilizes the overall system. For compensating *immediate reservation* scheme, *advance reservation* scheme has been introduced, for the client to make a reservation for resources for future resource usage. The client specifies time parameters to request advance reservation: START TIME and DURATION. Once admitted, the reserved resources will be effective after the start time for the duration. While those researches are well developed, they mostly treat *advance reservation* separately from *immediate reservation* so that they tend to give best-effort guarantees to *immediate reservation* and treat *advance reservation* as QoS guarantee scheme.

Without advance reservation, providing hard-QoS and soft-QoS to real-time (multimedia) applications running in a heterogeneous environment further complicates the problem of co-allocation. The objective of this section is to briefly touch upon the more important contributions that are directly relevant to this thesis. Therefore, this literature review

includes allocation and co-allocation schemes in a single environment as well as in a distributed environment.

4.2 Globus Architecture for Reservation and Allocation

There have been several attempts to solve the co-allocation problem in a Grid-like setting. Globus Architecture for Reservation and Allocation (GARA) system was proposed in [FoK99] to extend the Globus resource management architecture [CzF98]. The Globus system is a software infrastructure for sharing geographically distributed computational and information resources.

The Globus resource management architecture supports the co-allocation of heterogeneous compute resources to provide end-to-end computational QoS. Two issues that the Globus resource management architecture does not address: a) advance reservation which means that the required QoS cannot be guaranteed. Hence, the ability to perform co-allocation will be drastically restricted; and b) heterogeneous resource types. The absence of support for heterogonous resource types like network, disk, and others makes it impossible to provide end-to-end QoS guarantees when an application involves more than just computation [FoK99].

GARA extends this limitation by introducing the generic resource object where it reformulates computation-specific allocation functions in terms of general resource objects. This allows different application components to be manipulated in common

ways. GARA also introduces reservation, which will provide some confidence that subsequent allocation requests will succeed.

Since resources are independently controlled and administered, the GARA scheme will not work well because co-allocation requests can be rejected anytime in favor of internal requests. That is, there is no commitment from the GRAMs for the co-allocation classes. Furthermore, advance reservation is a requirement for co-allocation requests in GARA, which (as mentioned earlier) imposes strict timing constraints on the client side and does not allow the over subscription of the resources and thus leads to under utilization of the overall system.

4.3 Implicit Co-scheduling

Implicit co-scheduling [ArC98] is a new time-sharing approach for scheduling parallel applications that uses the communication and synchronization that occur naturally within the application to coordinate scheduling across workstations. Here, two events *response time* and *message arrival* are used to decide whether to continue with executing a subtask or to block it and schedule another subtask.

The basic idea is that, if a response to a request arrives, or a message arrives from a cooperating subtask executing on a different processor, it means that the remote subtask was scheduled at that time. Therefore, it is beneficial to continue executing the local subtask. On the other hand, if message arrivals do not occur, then the executing subtask

will use a two-phase spin blocking mechanism to wait. Under certain situations, waiting might be better than context switching to another subtask.

While implicit co-scheduling presents a new approach for improving the global performance of parallel applications, it falls short of addressing real-time multimedia applications. Further, unlike SQ, the implicit co-scheduling is targeted towards message passing subtasks. Implicit co-scheduling provides an application-level solution to the co-allocation problem (i.e., the application has to sense its own progress and adapt accordingly) whereas; SQ addresses the problem at the scheduler level. Thus, SQ does not require changes to the applications.

4.4 Scheduler for Multimedia And Real-Time applications

A proposed scheme in [NiL97], a scheduler for multimedia applications, Scheduler for Multimedia And Real-Time applications (SMART) is designed to support QoS sensitive and best effort applications as well as providing dynamic feedback to applications to allow them to adapt to the current load. In this way, SMART allows the user to prioritize across different classes of applications and dictate how resource(s) to be shared between applications with same priority. SMART is unique in regulating QoS sensitive tasks when the system is overloaded, while providing better value in under load conditions.

This suggests that SMART has a fairly complex resource management, which is fulfilled by basing the resource management decisions on two features; importance and urgency.

Importance is used to determine the overall resource allocation for each task. After the importance of each task is determined, SMART uses urgency based on earliest deadline scheduling to settle when each task is given its allocation.

Although SMART is effective in supporting multimedia application in a single system, it comes short of addressing the co-allocation problem in a Grid-like system, where resources are geographically distributed and administered by separate systems.

4.5 Proportional Share Algorithms

An alternative is to allocate each task some share of the CPU's capacity. Such algorithms are known as *proportional share algorithms* [WaW94, WaW95]. In these approaches, a machine will be shared by a real time scheduling policy and a conventional scheduling policy. An underlying proportional sharing mechanism will manage and therefore time slice between them. However, real time applications cannot effectively meet their deadlines as a result of conventional scheduler taking away resources at an inopportune and unexpected time in the name of fairness [NiL97].

The fundamental problem with proportional share or priority-based scheduling algorithms is the lack of *generality*. Proportional sharing is not suitable for QoS sensitive tasks because they cannot meet their time constraints effectively. With priority-based, QoS sensitive tasks are assigned higher priority than best-effort tasks. This is done, whether QoS sensitive tasks are important or not and causes all QoS sensitive tasks to run

ahead of best effort-tasks. To support such *generality*, hierarchical schedulers are proposed as a tool to support a variety of QoS sensitive as well as best effort applications. A local machine's resource (e.g. CPU) is partitioned among various application classes. This partitioning is referred to as *hierarchal partitioning* and will be explained in more detail in Section 5.6

4.6 Start-time Fair Queuing

The level of *generality* described above is achieved in Qlinux[1] by enabling hierarchical scheduling of applications and fairly allocating CPU bandwidth to individual applications and application classes. Qlinux accomplishes this by exploiting features such as Hierarchical Start-time Fair Queuing (H-SFQ) [GoG96].

Start-time Fair Queuing (SFQ) is a hierarchical scheduling algorithm that was proposed in [GoV96]. The work done by the CPU for a task is measured by the number instructions executed for that task. Then, the allocation of CPU is considered fair if, for all intervals $[t_1, t_2]$ in which two tasks are runnable, the difference of the normalized work (by the tasks' weight) received by them is as close to zero as possible.

Although SFQ supports different application classes, it works in a single environment. That is, it does not tackle the problem of resources co-allocation where a task requires different types of resources that are independently managed and reside on heterogeneous systems.

To conclude this section, the existing schemes to tackle allocation and/or co-allocation of resources are summarized. The schemes are presented along with their properties which can be classifies into 4 categories as explained at the end of section 3. The summary is provided in Table 1.

Table 1: Summary of existing allocation and co-allocation schemes.

Properties of Existing Allocation/Co-Allocation Schemes				Existing Scheme(s)
Supported application(s)	**Environment**	**Supported resource(s)**	**Used scheme**	
best-effort and QoS sensitive	Single	CPU	hierarchical	**H-SFQ**
best-effort and QoS sensitive	Single	CPU	hierarchical	**SMART**
best-effort	Single	diverse resources	proportional	**Lottery and Stride scheduling**
best-effort and QoS sensitive	distributed	diverse resources	hierarchical	**Tenet Suite 2**
best-effort	distributed	CPU	Application based	**Implicit co-scheduling**
best-effort and QoS sensitive	distributed	diverse resources	hierarchical	**GARA**

[1] Qlinux is a QoS enhanced Linux Kernel for Multimedia Computing: http//www.cs.umass.edu/~lass/software/qlinux/

5 SYNCHRONOUS QUEUING

5.1 Overview

Grid-based multimedia applications have a variety of QoS constraints that have to be met locally at each machine as well as globally across the Grid computing systems. A mechanism is needed to be positioned as a co-allocation infrastructure that will enable the usage of MEG environment resources for a variety of these challenging multimedia applications. The co-allocation issue that is addressed is concerned with ensuring that an application that has several sub components would be allocated sufficient resources so that all sub components of the application can make satisfactory progress with their execution. Some of these challenging multimedia applications include multimedia conferencing, virtual reality based distributed interactive simulation, distance learning, etc. The co-allocation is an essential feature for several important classes of multimedia applications. *Synchronous queuing* (SQ) is such a co-allocation scheme infrastructure that is capable of meeting those constraints for a satisfactory deployment of these applications. SQ algorithm is essentially a detection of asynchrony that can signal corrective action. Detection of asynchrony can be done at every schedule cycle or at a much larger interval (e.g. a group of schedule cycles). Corrective action is local to some extent and is done more often whereas global corrective action is done less frequently and is needed to handle heavy loading situations.

A detailed description and analysis of the SQ co-allocation scheme for MEGs are presented in this section. Discussion of the situations under which SQ can be used for

co-allocation in MEGs and the performance of the SQ co-allocation mechanism are also detailed.

5.2 Traditional QoS versus Synchronous Queuing QoS

SQ is a co-allocation scheme providing co-allocation with QoS constraints without advance reservation of resources. Hence, it is possible to perform co-allocation with hard QoS guarantees, as well as co-allocation with best-effort guarantees. That means all the local machines are capable of providing QoS which raises the following question: if there is a QoS guarantee from all the local machines, then a Grid-level QoS subtask s_i will be guaranteed its share of the local resource and thus, there is no need for either local or global synchronization and consequently there is no need for SQ.

To answer the above question, three points are considered; First, traditional QoS (admission control-based) guarantees are probabilistic in the sense that a subtask s_i requiring $m\%$ of a local machine's resource might get for each schedule cycle a different value x in the neighborhood of m depending on the machine's load. Note that, the application or task t has no control on the acceleration (upper) or retardation (lower) value of x. Second, traditional QoS guarantees are for the current schedule cycle and thus, it is an instantaneous guarantee and requires the application to be adaptive and sense its own progress. The traditional QoS scheme does not know or remember how much of the local machine's resource (e.g. CPU bandwidth) has been given to subtask s_i in the previous schedule cycles. Third, traditional QoS guarantees are environment-unaware

and do not know about other subtasks' QoS status in order to assure that the co-allocation skew is minimized for all subtasks belonging to the same task or application.

The above-mentioned points are fundamental differences between traditional QoS guarantees and SQ QoS guarantees. SQ is an aggregated scheme that assures the total work accomplished by each subtask s_i does not fall behind its agreed QoS. In addition, it is also an environment-aware scheme that assures the aggregated work accomplished by each subtask s_i does not fall behind the other subtasks belonging to task t.

5.3 Applications Suitable for Synchronous Queuing

Because the MEG is a specialization of the Grid concept to the multimedia applications, supporting wide ranges of applications is essential. In addition, classifying these challenging applications that might co-exist in the MEG environment is a key element in solving the co-allocation problem. In such a heterogeneous environment it becomes essential to subtasks $s_0,...,s_{n-1}$ (belonging to task t) to be synchronized in such a way that renders task t useful to the client or the end-user. This can be established by having the capability to assist each subtask to become environment-aware as well as providing aggregate rather than instantaneous QoS guarantees. Aggregate QoS guarantees provide the mean for controlling a subtask's consumption of a local machine's resources. The subtask is disciplined and accounted for the consumption since the time of execution up to the current schedule cycle. Environment-awareness is another essential capability for a subtask to have in such a distributed heterogeneous setting. This capability ensures

that a subtask is globally aware of the progress of all subtasks belonging to the same task t so that global synchronization can be signaled. To visualize these emerging synchronous-oriented MEG applications, consider the following example of a continuous multimedia application.

Consider a slide-show task/application t composed of two subtasks s_1 and s_2 responsible for displaying pictures and their corresponding text respectively. The co-allocation scheme used has to guarantee and monitor the QoS so that asynchronous situations (e.g. subtask s_1 is displaying a new picture, while subtask s_2 is still displaying the text of the previous picture or vise versa) are avoided as much as possible. This can be established by having the capability to assist each subtask to become environment-aware as well as providing aggregate rather than instantaneous QoS guarantees.

5.4 Queuing-based Architecture for Co-allocation

In a MEG environment, a QoS aware client, as shown in Figure 5, can submit a Grid QoS or a Grid best-effort task t simply by contacting a Grid resource broker. The client in this case is unable to execute task t locally due to lack of resources such as computing power, storage devices, etc.

Being part of a MEG environment, while cutting cost and time, allows sharing of resources that would otherwise be unavailable. Over the network, the Grid resource broker is client-Grid middleware that provides a uniform interface to heterogeneous

resources in conjunction with the Grid discovery and allocation services.

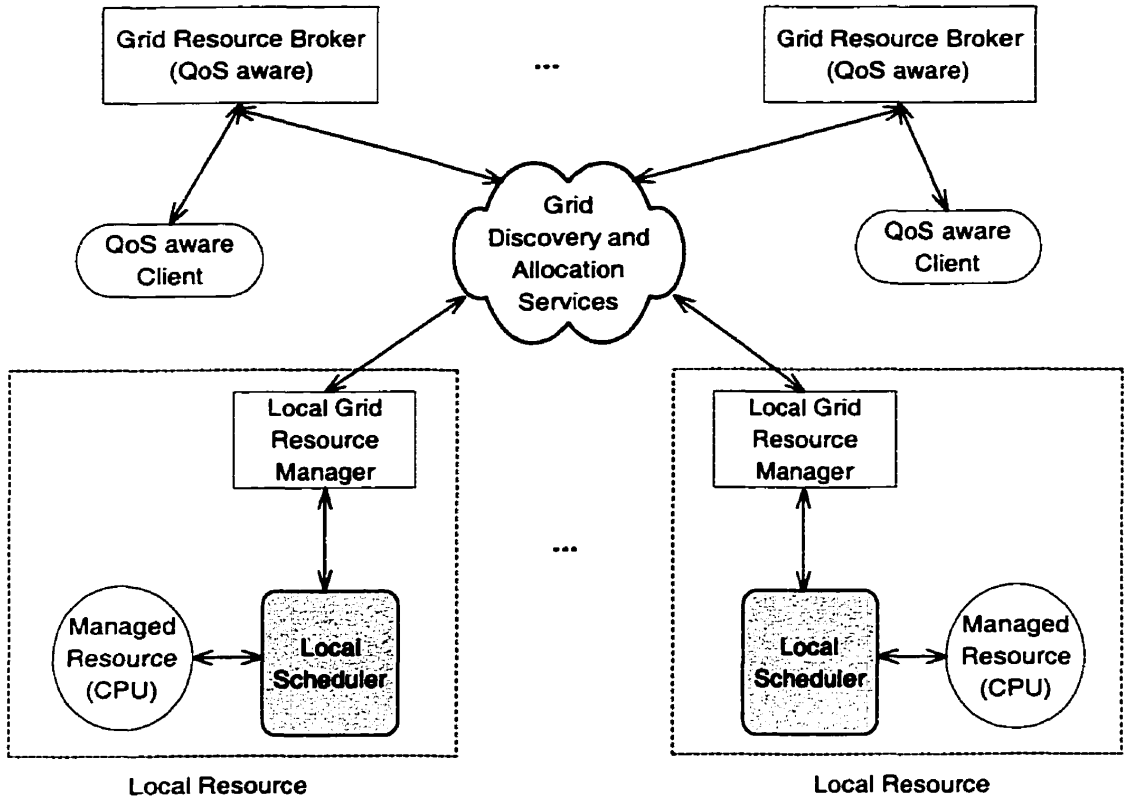


Figure 5: The overall queuing-based co-allocation architecture.

The Grid discovery and allocation services provides a bridge to the pool of available resources by constructing sets of resources that both match QoS requirements and conform to the local practices and policies of resource providers. Once the resources are discovered and allocated, the Grid-level scheduler, one service provided by the Grid discovery and allocation services, maps the allocation task t or the co-allocation task t's subtasks to the corresponding resource providers. The Grid controller, yet another service provided by the Grid discovery and allocation services, plays a crucial role in the SQ

co-allocation scheme by monitoring the progress of the different subtasks of task t and assuring that the QoS guarantee for task t is not violated. The local schedulers on the local resource providers or machines further schedule task t or its subtasks.

The local Grid resource manager, as shown in Figure 5, is a communication channel between the Grid discovery and allocation services and the local resources. One responsibility of the local Grid resource manager is to convey the local practices and policies of the local machine. Such practices and policies is partitioning of a local resource between local tasks, Grid QoS tasks, and Grid best-effort tasks. The local Grid resource manager is also responsible for monitoring and adaptively reporting the progress of the various tasks/subtasks executing on its local machine environment to the Grid controller. With limited local resource partition for each class (local, Grid QoS, and Grid best-effort) and depending on the QoS assurances sought, the local Grid resource manager may perform an admission test before admitting a task or a subtask. The Grid QoS tasks will have a mixture of tasks and subtasks some of which have hard QoS and others have soft QoS requirements. Providing QoS guarantees for these types of tasks or subtasks ensures that the requirements of admitted tasks and subtasks do not exceed the allocated resources assigned by the resources provider. Having strict admission control assures that the load of tasks/subtasks, in competing for a local resource, does not exceed the bandwidth of that resource. Traditional QoS has a strict admission control. In SQ, such strict admission control is relaxed in the sense that the load of admitted tasks/subtasks competing for a local resource could exceed the bandwidth of the local

resource. Hence, SQ will accommodate more tasks/subtasks and yet provide better QoS guarantees. The trades-offs that allow relaxed admission control are explained in detail in Section 5.10.

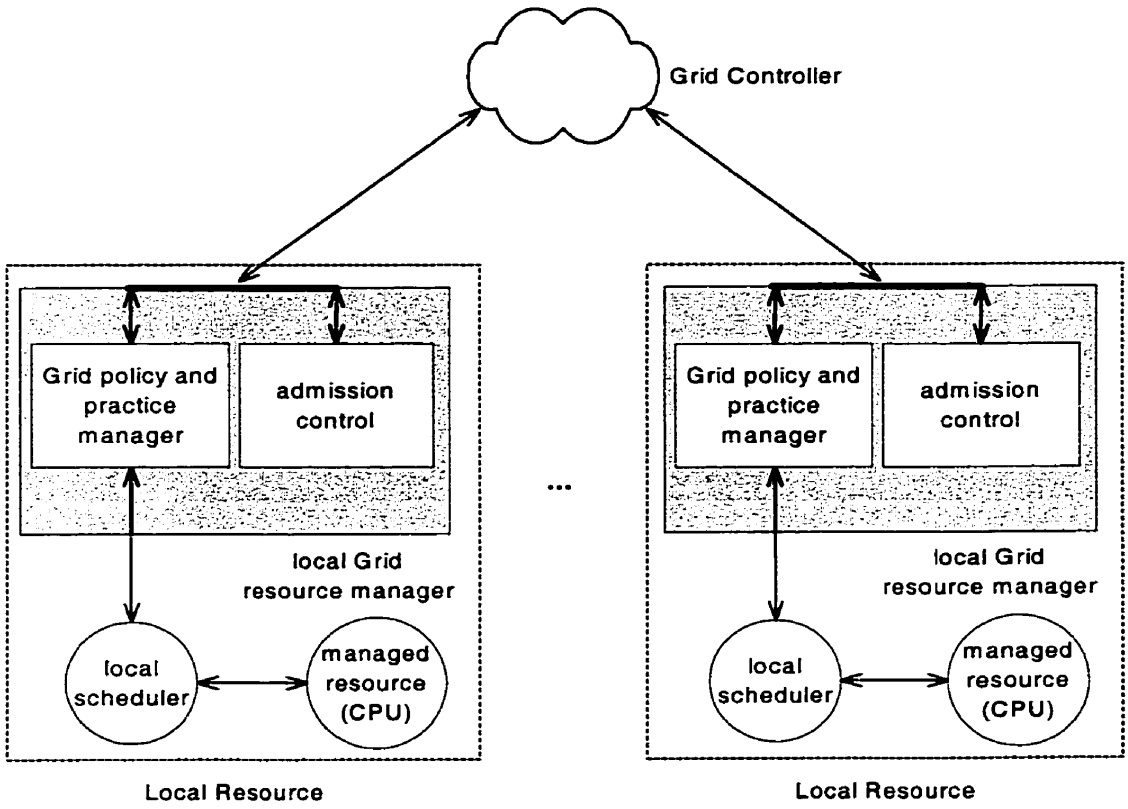


Figure 6: Architecture of the local resource.

The admission control unit as illustrated in Figure 6 performs the admission test. Once the admission control test is performed at the local node, it is the responsibility of the local Grid resource manager to convey the result, especially in case of rejection, to the Grid discovery and allocation services. The flow of tasks and subtasks coming to the local resource can be generated either locally or globally from the MEG environment. In turn the MEG traffic is further classified into two classes (Grid QoS and Grid best-effort),

as explained earlier. The hierarchical partitioning of a local managed resource (e.g. CPU) to accommodate this flow of tasks and the different components of the local scheduler are illustrated in Figure 7. The local scheduler can be viewed as the implementation of the policies and practices drawn by the Grid policy and practice manager. These implementations are used to manage and control local resources such as CPU.

5.5 Simplified Example

In this thesis, a queuing-based mechanism is presented to solve the co-allocation problem in MEGs. Unlike most of the previous approaches to co-allocation, this scheme does not require the target resources to support advance reservations. This allows for a flexible resource management scheme and also co-allocations with varying levels of QoS assurances. The basic idea is to adjust the resource allocations given to the different threads of the same application in an adaptive fashion so that the co-allocation skew is minimized among the threads that belong to the same application. Next, I sketch the overall synchronous queuing idea using a simplified example whereas subsequent sections will provide in detail SQ and the associated algorithms.

For brevity, consider that task t is subdivided into two subtasks s_0 and s_1. Let a_0 and a_1, with equal weights (i.e., $r_{a0} = r_{a1}$), be the two threads generated by the two local schedulers. The work done by a_0 and a_1 (i.e., the CPU quantum allocated to each) should be monitored to assure that aggregated work accomplished by each thread does not fall behind or exceed the other thread belonging to task t. One approach that can

be taken is to use real time (RT) and virtual time (VT) clocks [Zha91]. Let t_0 be the starting RT when a_0 starts execution. Initially $RT = pRT = VT = 0$, where pRT is the previous RT. For each schedule cycle (y), RT will be advanced by y. However, for the same schedule cycle, VT will be advanced by $VT + \frac{(RT - pRT)}{x} * x'$, where x is the agreed quantum allocated for a_0, and x' is the actual quantum a_0 gets. After VT is computed, pRT is set to RT.

Let us monitor thread a_0 after the j^{th} schedule cycle. For simplicity, let $VT = RT = pRT = 0$. Two scenarios can occur: first, a_0 is getting its quantum x in each schedule cycle, then $x = x'$ and hence VT will be advanced by $\frac{y}{x} * x' = \frac{y}{x} * x = y$. That is, in virtual time, the aggregate work done by a_0 is $\sum_{k=1}^{j} W_k^{a_0} = \sum_{k=1}^{j} y = j * y$ which is the same aggregate work expected in real time (RT). Second, a_0 is not getting its quantum x in each schedule cycle, then $x \neq x'$. Let us assume that a_0 is getting 90% of its agreed quantum (i.e., $x' = 90\% * x = 0.9x$). Hence VT will be advanced by $\frac{y}{x} * x' = \frac{y}{x} * 0.9x = 0.9y$. That is, in virtual time, the aggregate work done by a_0 is $\sum_{k=1}^{j} W_k^{a_0} = \sum_{k=1}^{j} 0.9y = j * 0.9y$ which is less than the aggregate work expected in real time (RT). If $j = 4$, then in virtual time, the aggregate work done by a_0 is $3.6y$, whereas the aggregate work expected in real time is $4y$. From the second scenario, we know

that a_0 is running behind its agreed schedule and some control mechanism has to be done. In this case, local synchronization is attempted to bring a_0 up to speed. If local synchronization fails, global synchronization is signaled. This is accomplished by sending messages to the other threads so that synchronization can be accomplished again for task t.

5.6 Tasks Flow Within Synchronous Queuing

Each local machine's load is a combination of the three flows of tasks; Grid QoS, Grid best-effort, and local tasks; which is assigned to the appropriate local queue waiting for execution as shown in Figure 7. A hierarchy of schedulers is used within each local scheduler.

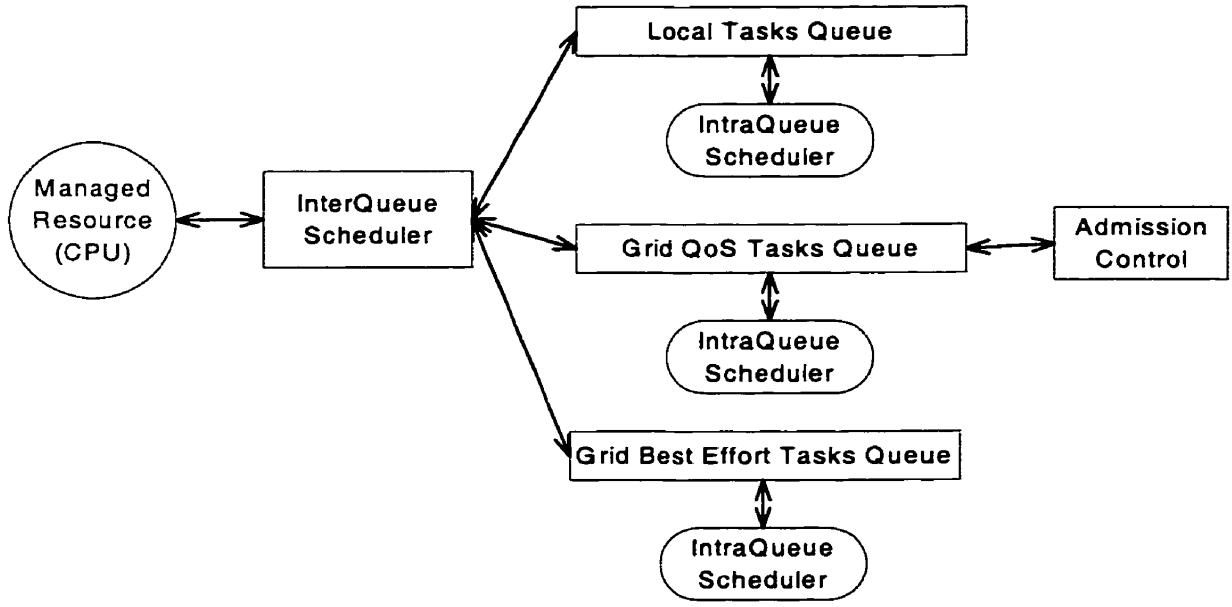


Figure 7: The different components of a local scheduler.

Each machine's resource (e.g. CPU) is hierarchically partitioned amongst the three task flows; local, Grid QoS, and Grid best effort. Statically assigned by the local resource, let r_{q1}, r_{q2} and r_{q3} denote the partition weights given to these class flows respectively such that $r_{q1}\% + r_{q2}\% + r_{q3}\% = 100\%$ of the machine's resource. As mentioned earlier, each flow will be assigned to its appropriate queue and hence each of the local queue (LQ), the Grid QoS queue (QoSQ), and the Grid best-effort queue (BEQ) will have associated weights of r_{q1}, r_{q2} and r_{q3} respectively.

Locally generated tasks require allocation of local resources and are assigned to the LQ. As a resource provider to MEG, a local machine is expected to accommodate Grid flow tasks or subtasks as well. A Grid task may have hard QoS, soft QoS, or best-effort requirements. Some of these Grid QoS tasks and subtasks might have co-allocation requirements and others may not. The Grid QoS tasks and subtasks are assigned to the QoSQ while the Grid best-effort tasks and subtasks are assigned to the BEQ. The interQueue scheduler determines which queue should be selected whereas the intraQueue scheduler decides which task or subtask should be scheduled from the selected queue.

5.6.1 Grid Task's Weight Assignment

When the Grid-level scheduler assigns Grid tasks or subtasks, a standard CPU speed of one GHz is assumed. Let t denote a Grid QoS task that requires $m\%$ of CPU bandwidth and let this task t be composed of n subtasks $s_0,...,s_{n-1}$. Furthermore, let $a_0, a_1,..., a_{n-1}$ be the threads that are instantiated at the local machines for the subtasks $s_0,...,s_{n-1}$,

respectively. Each of these threads will have a weight, r_{ai}, assigned by the Grid-level scheduler. Thus, each thread will be asking for $m_{ai}\%$ of the local machine's CPU bandwidth given by the following equation $m_{ai} = \left(\frac{d * m * r_{ai}}{r_a} \right)$, for all $0 \le i \le (n-1)$, where $d = \frac{1GHz}{CPU_l}$, i.e. d is the standard machine CPU speed divided by the local machine's CPU speed, and $r_a = \sum_{i=0}^{n-1} r_{ai}$, i.e. r_a is the sum of the weights for all the subtasks that belong to task t.

Now, m_{ai} is calculated based on 100% of CPU availability, so we have to map it to r_{q2} (partition weight of CPU allocated to the Grid QoS flow). This is accomplished using the following scaling $m_{ai} \Leftarrow \frac{r_{q2} * m_{ai}}{100}$.

As an illustrated example, let 100 CPU quanta represent a schedule cycle and consider a Grid QoS level task t asking for 5% of a resource (e.g. CPU) every schedule cycle and specifically 5 quanta of the local CPU every schedule cycle on a standard machine running at 1 GHz. Suppose that task t is composed of 4 subtasks $s_0, s_1, s_2,$ and s_3. Furthermore, let $a_0, a_1, a_2,$ and a_3 be the threads that are instantiated at the local machines for the subtasks $s_0, s_1, s_2,$ and s_3 respectively. Each of these threads will have a weight, r_{ai}, assigned by the Grid-level scheduler as shown in Figure 8. Let thread a_0 be

given a weight r_{a0} of 1 and assigned to a local machine running at 100 MHz and let $r_a = 10$. Thus, thread a_0 will be asking for m_{a0} share of its local machine given by as follows: $m_{a0} = \left(\frac{1}{10} * 5 * \frac{1{,}000{,}000{,}000}{100{,}000{,}000} \right) = 5$. Therefore, thread a_0 requires 5 CPU quanta every schedule cycle on this local machine. Now, keep in mind that the resource (CPU) of the local machine is partitioned amongst three task classes, so thread a_0 CPU requirement needs to be mapped to the QoSQ weight, which is r_{q2}. If we let $r_{q2} = 40\%$, then the weights for thread a_0 can be mapped as follows: $m_{a0} = \left(\frac{40}{100} * 5 \right) = 2$ and thus a_0 is asking for 2 CPU quanta every schedule cycle on its local machine. In a similar fashion, the weights for the other threads a_1, a_2, and a_3 can be computed following the same procedure.

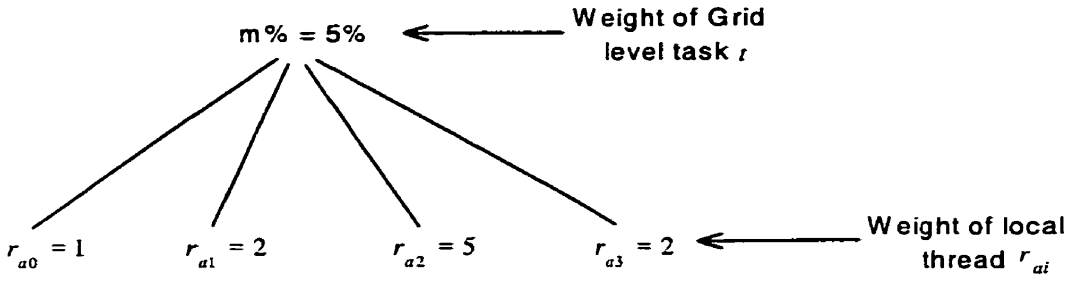


Figure 8: Assigning weights for a Grid level task and its four local subtasks.

5.7 Hierarchy of Local Schedulers

The interQueue scheduler shown in Figure 7 uses SFQ [GoV96], which enables the co-existing of resource allocation algorithms, achieves fair resource allocation among the three local queues, and requires only relative importance of tasks expressed by weights to be known. SFQ achieves CPU fairness allocation amongst the threads based on their associated weights.

The objective of SFQ is to allocate CPU quantum/quanta to threads proportional to their weight. To achieve this objective, SFQ assigns a *start tag,* and *finish tag* to each thread and also assigns a common *virtual time.* SFQ schedules the threads in the increasing order of start tags and ties are broken arbitrarily. *Start tags, finish tags,* and *virtual time* are initially 0. When the CPU is idle, the *virtual time* is set to the maximum of *finish tag* assigned to any thread. On the other hand, when the CPU is not idle, *virtual time* is set to the *start tag* of the queue in service. When the scheduling quantum for the thread finishes execution, two things happen:

- the thread's *finish tag* is incremented by the following equation:

$$finishtag = starttag + \left(\frac{b}{r_{ai}} \right), \text{ where } b \text{ is the length of the scheduling quantum}$$

  for thread a_i, and r_{ai} is the weight for thread a_i.

- the thread's *start tag* is computed as the maximum of the *virtual time* or its *finish tag*

Since each machine's local resource (e.g. CPU) is hierarchically partitioned amongst the

three queues; LQ, QoSQ, and BEQ, the objective of SFQ is to assure that the allocation of machine's local resource to the three queues is proportional to their respective associated weights (r_{q1}, r_{q2} and r_{q3}). The machine's local resource of each machine is statically partitioned amongst the three queues (i.e. flows of tasks/subtasks). The pseudo-code of the interQueue selection scheme is presented in Figure 9.

```
// queue is the queue that will be selected by the interQueue scheduler.
// LQ is the local queue.
// QoSQ is the Grid QoS queue.
// BEQ is the Grid best effort queue.
SelectQueue()
{
        queue = FindMinStartTag(LQ, QoSQ, BEQ)

        if ( queue is empty ) //equivalent to CPU idle
                virtual time = FindMaxFinishTag(LQ, QoSQ, BEQ)
        else
                virtual time = queue start tag
        endif

        queue finish tag = queue start tag + ( scheduling quantum length / queue weight )
        queue start tag = max(queue finish tag, virtual time)

        return(queue)
}
```

Figure 9: The interQueue SFQ pseudo-code for selecting a queue.

As an illustrated example of how SFQ works, assume that the local queues LQ, QoSQ, and BEQ are given 40, 40, and 20 weights respectively. Each local queue will be given *starttag* and *finishtag*. Furthermore, a common *virtual time* will be assigned. Initially, *starttag*, *finishtag*, and *virtual time* are all set to zero. Let the scheduling quantum length

for each queue be one second. Since ties are broken arbitrarily, assume that LQ is scheduled first. Since, *virtual time* is defined to be the *starttag* of the thread in service, *virtual time* is set to zero and the *finishtag* for LQ is set to $0+\frac{1}{40}=0.025$. In addition, the *starttag* of LQ is set to $\max\{0,0.025\}=0.025$. At this time, SFQ will schedule QoSQ or BEQ because their *starttags* are smaller that LQ's *starttag*. In the same manner SFQ will continue to schedule these three queues and if we complete this for 10 scheduling quanta, we will find that the 10 scheduling quanta is proportionally assigned as 4, 4, 2 to LQ, QoSQ, and BEQ respectively.

After determining the queue to schedule next, the intraQueue scheduler of the selected queue determines which task or subtask should be executed from the selected queue. Depending on the service offered by each queue, a particular scheduling algorithm for the selected queue is exploited. As the intraQueue scheduler, *round robin* (RR) scheduler is used for the LQ and BEQ whereas SFQ is used for QoSQ because fairness in resource allocation is sought in order to provide QoS guarantees.

RR is designed especially for time-sharing systems. A small unit of time, called timeslice or quantum, is defined. All tasks/subtasks are kept in a circular queue. The CPU scheduler goes around this queue, allocating the CPU to each task/subtask for a time interval of one quantum. New tasks/subtasks are added to the tail of the queue. The CPU scheduler picks the first task/subtask from the front of the queue, sets a timer to interrupt after one quantum, and dispatches the task/subtask. If the task/subtask is still running

```
// queue is the queue selected by the interQueue scheduler.
// intraAlgorithm is the scheduling algorithm (e.g. round robin, SFQ) that is used by
// the intraQueue scheduler to schedule a task or subtask from the selected queue.

ScheduleTask ()
{
        // the tasks and subtasks are being generated by another process and
        // assigned to one of the three different queues accordingly.
        // SelectQueue() is defined in Figure 8.
        queue = SelectQueue ()

        if (queue.id == QoS)
                intraAlgorithm = SFQ
        else
                intraAlgorithm = RR
        endif

        task = SelectTask (queue, intraAlgorithm)

        //start to execute the task
        // schedule the task and put it back in the queue

        if (task execution time > machine service time)
                        task execution time = task execution time – machine service time
                        increment the machine real time by machine service time
                        enqueue(task, queue)

        // schedule the task and remove it from the queue
        else
                        task execution time = 0
                        increment the machine real time by machine service time
                        dequeue(task, queue)
        end if
}
```

Figure 10: Pseudo-code for selecting and executing a task.

at the end of the quantum, the CPU is preempted and the process is added to the tail of the queue. If the task/subtask finishes before the end of the quantum, the process itself

releases the CPU voluntarily. In either case, the CPU scheduler assigns the CPU to the next task/subtask in the ready queue. Every time a task/subtask is granted the CPU, a context switch occurs, which adds overhead to the task/subtask execution time.

In the simulation, the context switch overhead is neglected. Each time a task/subtask is scheduled, the task/subtask's execution time is subtracted by the machine service time while the machine real time is incremented by the machine service time. If the scheduled task/subtask's execution time is less than or equal to the machines service time, then its execution time is set to zero and it is removed from the queue. The pseudo-code to schedule tasks and subtasks is presented in Figure 10.

5.8 Strict Versus Relaxed Admission Control

In traditional QoS admission-based algorithms, QoS is provided by having strict admission control assuring the load competing for a local machine's resource does not exceed the upper limit availability of the local resource. Hence, the system will never be overloaded and the following condition will hold true $w - r_{q2} \geq 0$, where $w = \sum_{i=0}^{n-1} r_i$, i.e. w is the sum of the weights for all the tasks and subtasks that are in QoSQ and competing for the local resource, and is the available weight associated with the QoSQ. Having strict admission control does not assure QoS guarantees especially under situations where $w - r_{q2} > 0$ meaning that the local machine in under loaded. In this case, using a scheduler such that SFQ will assure that each task and subtask gets its share

of the local resource (e.g. CPU) proportional to its weight. So, if the machine is under loaded, tasks or subtasks will get more than their agreed share of the local resource (i.e. task's virtual time > task's real-time) and co-allocation skew situations will occur.

Relaxed admission control accommodates more QoS demands of a local resource than is provided by the resource provider. In this case, the system can be overloaded $w - r_{q2} < 0$) or under loaded ($w - r_{q2} > 0$). So, the probability of having co-allocation skew is higher with relaxed admission control. In spite of that, SQ scheme, which uses relaxed admission control, outperforms strict admission control-based scheme by a significant margin (refer to Section 6). For the under loaded situation, the scenario will be as explained with strict admission control. With the overloaded situation, we have $w > r_{q2}$ and some of the subtasks will be getting less than the agreed weight causing a co-allocation skew to occur.

5.9 Basic SQ Co-allocation Algorithm

After each schedule cycle (monitor cycle) or a much larger interval (e.g. a group of schedule cycles), the local scheduler, through the Grid policy and practice manager, reports the progress of the co-allocation subtasks to the Grid controller. For each schedule cycle (y), real time (RT) is advanced by y and the local scheduler calculates virtual time (VT) for each of its subtasks. As explained in section 5.5, the calculation of virtual time will be by following the equation: $VT + \frac{(RT - pRT)}{x} * x'$, where x is the

agreed quantum allocated for s_0, and x' is the actual quantum s_0 gets. It can be noticed that as x' approaches x, VT approaches RT and hence the finishing time for the subtask is approaching the expected finish time. Once, the Grid controller receives the subtasks' execution progress report from the local schedules, it calculates a pivotal point for each task and then performs detection of asynchrony test. Upon the outcome of this test, the Grid controller might take a corrective action.

SQ is basically applied to hard QoS tasks for which missing a deadline leads to catastrophic failures. These applications require a deterministic guarantee for their QoS parameters and thus the Grid controller has to adaptively monitor their execution and accordingly signal the appropriate corrective action. The next few subsections discuss in detail the steps taken to perform the SQ co-allocation scheme. These steps involve selection of a pivotal point, performing the detection of asynchrony, and signaling the corrective action. Detection of asynchrony involves performing the asynchrony and the overall deviation tests. The overall deviation test can be further classified into two steps: overall retardation test and overall acceleration test.

5.9.1 Selecting a Pivotal Point

Upon receiving the information on the progress of the co-allocation subtasks from the local machines, the Grid controller selects a pivotal point (pp) that is calculated as follows:

$pp = \frac{\sum_{i=0}^{n-1} VT_i}{n}$, where n is the number of subtasks belonging to task t, and VT_i is the virtual time for subtask s_i. So, the pivotal point is essentially the average of virtual time for the n subtasks that belong to task t.

5.9.2 Detection of Asynchrony

For each Grid QoS task, the clients have to provide two QoS attributes: *asynchrony*, and *overall deviation*. *Asynchrony* is the acceptable *async* that a task t can tolerate and is calculated as: $async = VT_f - VT_s$, where VT_f is the virtual time of the fastest subtask, and VT_s is the virtual time of the slowest subtask among all subtasks belonging to task t. O*verall deviation* is the acceptable retardation or acceleration that a task t can tolerate for its subtasks. Retardation puts a lower bound on how much a subtask's virtual time can be behind its real time, whereas acceleration puts an upper bound on how much a subtask's virtual time can be a head of its real time.

Suppose that a task t composed of 5 subtasks $s_0,...,s_4$, where d_i represents the deviation of a subtask from its expected finish time. The *asynchrony* window and the *overall deviation* window are illustrated in Figure 11.

For each task t, its pivotal point is checked whether it falls within the *overall deviation* window. As shown in Figure 12, the outcome from the overall deviation test can be one

of the following:

- Yes, the pivotal point falls within the *overall deviation* window. If this is the case, then the asynchrony test is performed to assure that *Async* (the difference between virtual time of the fastest subtask and the virtual time of the slowest subtask among all subtasks belonging to task t) is within the asynchrony window.
- No, the pivotal point falls outside of the *overall deviation* window. In this case or in the case where the asynchrony test fails, corrective action is required.

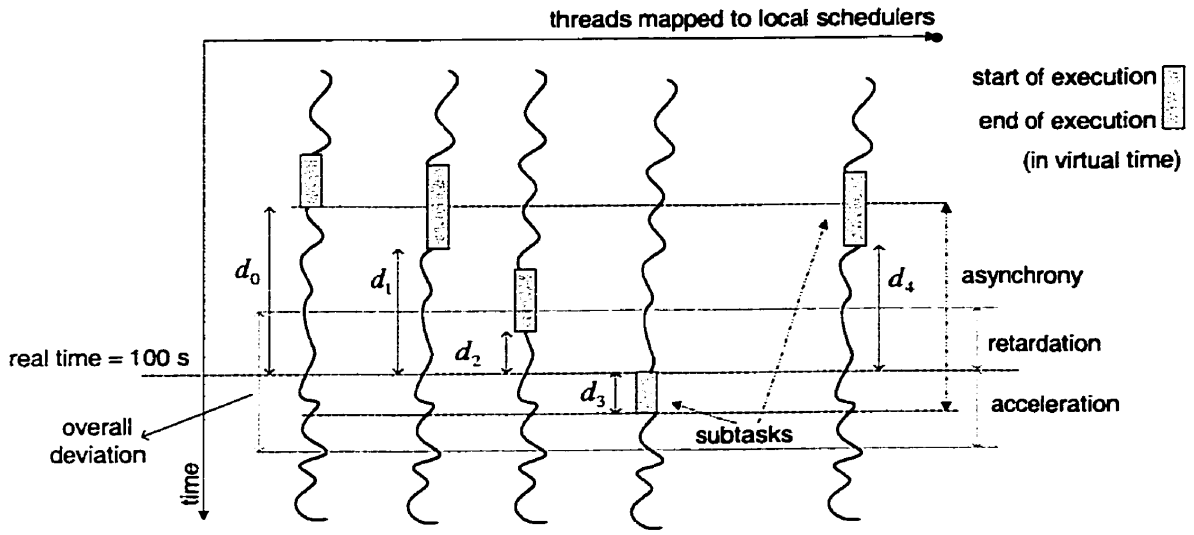


Figure 11: Progress of subtasks in the first schedule cycle.

```
// The Grid controller executes this code.
// queue containing all subtasks belonging to a hard QoS task t.
// QoSQ is the queue containing all hard and soft QoS tasks and subtasks.
// async is as defined in subsection 5.8.2 by the equation  async = VT_f - VT_s

Monitor(queue) {
        while (QoSQ is not empty)
                //dequeue all subtasks belonging to a hard QoS task t
                queue = dequeue(QoSQ)

                //calculate pivotal point of task t where its subtasks are in queue
                pp = calculate_pp(queue)

                if (pp is within the overall window)
                        if (async > asynchrony window)
                                corrective_action(queue)
                        endif
                else
                        corrective_action(queue)
                endif
        endwhile
}
```

Figure 12: Pseudo-code for detection of asynchrony.

5.9.3 Corrective Action

At this point, the Grid controller (based on information collected globally) signals a local machine for a corrective action. The corrective action can be to speedup or slowdown a

subtask s_i. The local machine might succeed or fail in carrying out the corrective action locally.

Failure can happen in situations where subtask s_i needs to speed up and the local machine is overloaded. In other words, the local machine has no extra CPU quanta to spare. In this case, the local machine reports back to the Grid controller for a global corrective action to take place.

On the other hand, success can happen in situations where subtask s_i needs to slow down, which means that the local machine subtask s_i is running on is under loaded. In this case there are extra CPU quanta that are given to subtask s_i. One way, the extra CPU quanta can be absorbed is to create *an idle* task and assigns it a weight equal to the extra CPU quanta. Care has to be taken of whether to penalize subtask s_i and lower its weight to compensate for the extra CPU quanta it absorbed.

Since SQ is an ongoing feedback process, its effectiveness might take a few schedule cycles before satisfying the QoS attributes given by the client (*asynchrony*, and *overall deviation*). Figure 13 presents the monitor module within SQ the co-allocation scheme.

5.10 Isolation Guarantee

By using relaxed admission control, SQ admits more load (tasks/subtasks competing for a

local resource) than the available local resource bandwidth. As mentioned at the end of Section 5.6, Grid QoS task may have hard QoS or soft QoS requirements. Allowing more demand than what is available (i.e. implementing relaxed admission control) assures that some of the QoS tasks/subtasks will be getting less CPU quantum/quanta than what they expressed in their weights. Since missing a deadline for hard QoS tasks/subtask will result in a catastrophic failure, the trade off that SQ makes is to borrow the needed CPU quantum/quanta from soft QoS tasks/subtasks by reducing their weights and lending the borrowed weights to the needy hard QoS tasks/subtask.

```
// The Grid controller executes this code.
// queue containing all subtasks belonging to a hard QoS task t.
// subtask is one of the subtasks of the hard QoS task t.
corrective_action(queue) {
        while (queue is not empty)
                //dequeue a subtask
                subtask = dequeue(queue)
                //determine the appropriate action to be taken.
                // the action can be speeding up or slowing down the subtask
                action = determine_action(subtask)
                //signal the subtask's local machine to carry the action
                //if the action can not be carried locally
                        //mark this subtask's action to be carried globally
        endwhile
}
```

Figure 13: The Global controller corrective action module in SQ.

A needy hard QoS task/subtask will have its weight increased and this will not affect any other hard QoS tasks/subtask because their weights are not affected and hence SFQ will assure their share of the CPU remains the same. Therefore, SQ guarantees a total isolation between the hard QoS tasks/subtasks.

Furthermore, whatever happens (increasing or decreasing tasks/subtasks' weights) in QoSQ does not affect the other two queues (LQ and BEQ) because each weight associated with LQ and BEQ is not affected and thus the interQueue scheduler (SFQ) assures LQ and BEQ their share of CPU remains the same. In conclusion, SQ guarantees a total isolation between the tasks/subtasks in QoSQ as well as a total isolation between the three different queues (LQ, QoSQ, and BEQ).

5.11 Scheduling Concepts With SQ

Introducing the hierarchy of schedulers such as interQueue and intraQueue schedulers SQ uses in a new concept introduced. The concept of *real* and *virtual* time [Zha91] is used by Lixia Zhang as a data traffic control in high-speed networks. The concept is used by SQ but is extensively altered to account for the previous work done by each task/subtask so SQ can provide an aggregated work. Also, the scheme of calculating the *co-allocation skew* detailed in Section 5.9 is original with this thesis

6 SIMULATION RESULTS AND DISCUSSION

6.1 Overview

The Grid topology model used in the simulation is discussed in detail in section 6.2. The simulation model is written using Java base classes [ArN99] and further extensively modified to fit our purpose. The effectiveness and performance of SQ are assessed by writing a discrete event simulation modeling the Grid topology shown in Figure 14. The proposed SQ algorithm for synchronizing multimedia applications was simulated using the Advanced Networking Research Laboratory (ANRL) facilities.

The performance of SQ was compared to the traditional QoS with strict admission control to assess the advantages and to show the benefits of SQ. The next sections describe the simulation model, the performance measures, and parameters used in the simulation. This is followed by the simulation results and discussions.

6.2 Goals of The Simulation

The goal of the simulation is to investigate and examine the co-allocation problem. The co-allocation is defined as simultaneous allocation of resources to subtasks belonging to a task running on geographically distributed machines. The goal of the simulation is to look into and focus on the co-allocation problem and explore the effectiveness of SQ in reducing (minimizing) the co-allocation skew among the different subtasks belonging to task t.

Let us now think about how the co-allocation skew occur? Let task t composed of 2 subtasks s_0 and s_1. Furthermore, let s_0 be asking for 2 and s_1 be asking for 1 CPU quanta every schedule cycle. The co-allocation skew occurs when the scheduler start giving the subtasks CPU quanta different from what they asked for. But if, for every schedule cycle for the life time of the subtasks, the scheduler gives 2 and 1 CPU quanta to subtask s_0 and s_1 respectively, there will be no co-allocation skew and both of the subtasks will be executing in synchronization. This is the optimal scenario, but in real system this is not the case because machines can be underloaded or overload? If the machine is underloaded, then the subtasks will get more than what they asked for, and if the machine is overloaded the tasks will get less than what they asked for. In both cases, co-allocation problem will occur. Underloaded situations can happen with having strict admission control, but for overloaded situations one might ask: QoS is provided by having admission control. Hence, the system will never be overloaded and the following condition will hold true $w - r_{q2} \geq 0$, where $w = \sum_{i=0}^{n-1} r_i$, i.e. w is the sum of the weights for all the tasks and subtasks in QoSQ competing for the local resource, and is the available weight associated with the QoSQ. The answer to such question is yes. This is true if strict admission control is used, but SQ uses relaxed admission where the situation $w - r_{q2} < 0$ will occur for sure because we are admitting more total CPU demand (expressed in tasks/subtask weight) for a local resource than the availability of CPU cycles at hand. Hence, the probability of co-allocation skew situations will occur more often for SQ than a strict admission control-scheme.

Now, let us think about what contributes to the complexity of the co-allocation skew? That is what makes it a harder problem to manage? When we look at a co-allocation skew, the number of subtasks plays the primary role in the complexity of the problem. In the simulation study, I looked at situations where co-allocation skew is somewhat an easy problem to manage and at the same time I compared and investigated the situations where the co-allocation skew problem is much harder. In particular, I increased the offered load with the increased number of local machines. What can be learned from this as opposed to a study in which job resource requirement is fixed while number of machines is increased? I can purely test the effectiveness of SQ as well as QoS schemes as the co-allocation gets harder and harder to manage (i.e. as the number of subtasks increase).

6.3 Simulation Model

The Grid topology model used in the simulation is shown in Figure 14. Each tasks generator model generates a Poisson stream of tasks/subtasks with specified Mean InterArrival Time (λ) until a specific number of tasks have been generated. The local machines are heterogeneous and each reports the execution progress of its co-allocation subtasks to the Grid controller. Each machine has a local generator generating best effort local tasks and assigns them to the local queue (LQ). Two global generators are at the Grid level and they are responsible of generating Grid QoS and Grid best effort tasks/subtasks. The global tasks/subtasks are assigned to the Grid QoS queue (QoSQ) and the Grid best effort queue (BEQ) accordingly. Furthermore, the Grid QoS tasks are stochastically divided into hard and soft QoS tasks.

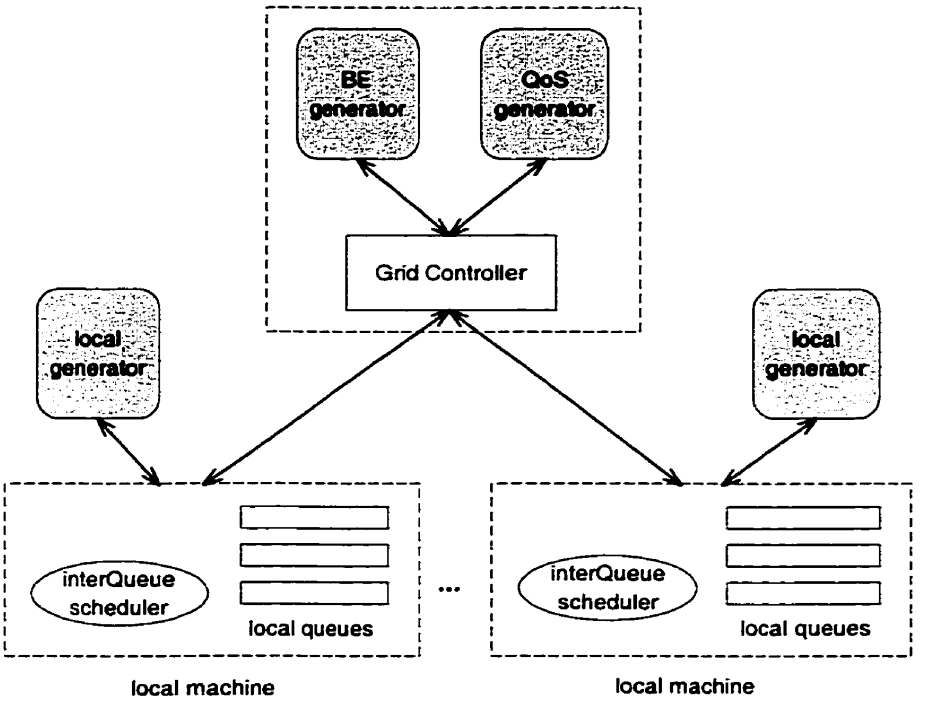


Figure 14: Grid topology used in the simulation.

To sufficiently assess the performance of SQ co-allocation algorithm, various performance metrics need to be explored. Even though the co-allocation skew is one of the quantities that the SQ algorithm is trying to minimize, the trade-offs and their impact on the system should be examined. The trade-offs that are examined in the simulation are acceptance ratio, QoS conformance, and effective machine usage.

The four performance metrics used to assess the performance of SQ are acceptance ratio, effective machine usage, QoS conformance, and average co-allocation skew and they are

defined as:

$$\text{acceptance ratio} = \frac{\text{number of QoS tasks accepted}}{\text{total number of QoS tasks generated}}$$

$$\text{effective machine usage} = \frac{\text{number of hard QoS tasks conforming to } \textit{asynchrony} \text{ window}}{\text{total number of hard QoS tasks accepted}}$$

$$\text{QoS conformance} = \frac{\text{number of hard QoS tasks confirming to } \textit{overall deviation} \text{ window}}{\text{total number of hard QoS tasks accepted}}$$

$$\text{average co-allocation skew} = \frac{\sum_{i=1}^{i=m}\left(\sum_{j=0}^{j=n-2}\left(VT_{ij} - VT_{is}\right)/(n-1)\right)}{\text{total number of hard QoS tasks accepted}}$$, where VT_{ij}

is the virtual time of subtask j of task i, and VT_{is} is the virtual time of the slowest subtask belonging to task i.

The term randomly generated over a range [a, b] means that the number is generated using a discrete (integer-valued) uniform distribution over $a, a+1, \ldots, b$ inclusive. That is written as U[a, b]. The Grid topology used in the simulation consists of local machines (*Nloc*) set deterministically at [5,10,15, 25] and 3 generators each generating tasks (t) randomly generated over a range [1000, 2000]. For each simulation run, the generators generate a Poisson stream of tasks with specified λ set deterministically at [10, 100, 200, 500] seconds. For each QoS task, the two QoS attributes provided by the user are *asynchrony* and *overall deviation*, which are randomly generated over a range [100, 500] seconds. Furthermore, each Grid task is composed of subtasks (n) randomly

generated over a range [1, # of local machines] and each of these subtasks is assigned an execution time ( ) randomly generated over a range [1500, 2000] seconds. The value is chosen to be large enough to resemble a continuous media application so SQ (being a feed back scheme) will have enough time to "kick in" and carry its corrective action and hence be effective. The CPU speed for each local machine (*LCPU*) is randomly generated over a range [100, 600] MHz. A Grid level CPU bandwidth (*GCPU*) of one GHz is assumed when assigning Grid tasks/subtasks to local machines as explained in section 5.6.1 and the CPU bandwidth of each local machine is statically partitioned among the 3 flows of tasks/subtasks. Furthermore, a weight (m) is assigned to a Grid QoS task t and a weight (r_{ai}) is assigned to a thread representing subtask s_i belonging to task t (refer to Subsection 5.6.1). Also, r_a is the sum of the weights for all the subtasks belonging to task t. The two weights m, and r_{ai} are randomly generated over a range [1,5] of CPU quanta. For the sake of computing mean value analysis, I will refer to m, r_{ai}, and r_a as representing the mean value of their associated weights respectively. Table 2 and Table 3 show the design and exogenous parameters used in the simulation. In addition, Table 4 shows the two algorithms and their parameters used in the simulation.

Table 2: Design parameters used in the simulation.

Symbol	Definition	Design Parameters Values
λ	mean inter-arrival time (second)	$\lambda = (10,100,200,500)$
Nloc	Number of machines	*Nloc* = (5,10,15,25)
GCPU	the assumed Grid level standard CPU speed	*GCPU* = 1 *GHz*
Reps	how many times the simulation run is repeated for each point in the graphs	Re *ps* = 50

Table 3: Exogenous parameters used in the simulation.

Label	Definition	Exogenous Parameters Distribution
assynchrony	QoS attribute specified by user (second)	*assynchrony* = U[100,500]
overall deviation	QoS attribute specified by user (second)	*overall deviation* = U[100,500]
n	Number of subtasks	$n = U[1, Nloc]$
	Number of the execution time (second)	$\mu = U$[1500,2000]
m	task's weight (CPU quanta)	$m = U$[1,5]
r_{ai}	Subtask's weight (CPU quanta)	$r_{ai} = U$[1,5]
LCPU	CPU speed of local machine	*LCPU* = U[100,600] *MHz*
t	Number of tasks	$t = U$[1000,2000]

Table 4: Different classes of algorithms used in the simulation.

Parameter	Algorithm Used	
	Traditional QoS guarantees	SQ
IntraAlgorithm	SFQ	SFQ
Admission Control	Strict	Relaxed
Queue Used	Grid QoS	Grid QoS
Performance Metric Used	All	All

6.4 Mean Value Utilization Analysis

In this subsection, mean value utilization analysis is performed to compute total resource demand and compare it to available resources (CPU cycles). After that I relate to admission control to help in explaining the results of the simulation. For each entry in Table 5, ρ is calculated as *total CPU demand* over *CPU cycles available. Total CPU demand* and *CPU cycles available* are computed over an interval of 100 seconds.

***Total CPU demand* is calculated as:**

mean number of tasks generated every 100 seconds * n * mean subtask's weight. Tasks generated every 100 seconds $= \frac{100}{\lambda}$ and mean subtask's weight (based on 100% CPU availability) $= \frac{r_{ai} * m * GCPU}{r_a * LCPU}$, where $r_a = n * r_{ai}$

Therefore, *total CPU demand* is given by $\frac{100}{\lambda} * n * \frac{r_{ai} * m * GCPU}{(n * r_{ai}) * LCPU}$. Furthermore, *total CPU demand* (based on the weight associated with QoSQ which is r_{q2}) is $\frac{100}{\lambda} * n * \frac{r_{ai} * m * GCPU}{(n * r_{ai}) * LCPU} * \frac{r_{q2}}{100} = \frac{100}{\lambda} * \frac{m * GCPU}{LCPU} * \frac{r_{q2}}{100}$. Refer to subsection 5.6.1. for more detail.

***CPU cycles available* is calculated as:**

CPU cycles available every 100 seconds for QoSQ $* Nloc = r_{q2} * Nloc$.

Finally, $\rho = \left(\frac{100}{\lambda} * \frac{m * GCPU}{LCPU} * \frac{r_{q2}}{100} \Big/ r_{q2} * Nloc \right)$. Since $GCPU$, $LCPU$, m, and r_{q2} are constants, the above equation can be simplified as follows:

$$\rho = \left(\frac{100}{\lambda} * \frac{3 * 1{,}000{,}000{,}000}{350{,}000{,}000} * \frac{40}{100} \Big/ 40 * Nloc \right) = \frac{60}{7 * \lambda * Nloc}.$$

Table 5: Mean value utilization of the different number of machines as λ increase.

λ	Number of machines			
	5	10	15	25
10	$\rho = 17.1\%$	$\rho = 8.57\%$	$\rho = 5.71\%$	$\rho = 3.43\%$
100	$\rho = 1.71\%$	$\rho = 0.86\%$	$\rho = 0.57\%$	$\rho = 0.34\%$
200	$\rho = 0.86\%$	$\rho = .43\%$	$\rho = 0.29\%$	$\rho = 0.17\%$
500	$\rho = 0.34\%$	$\rho = 0.17\%$	$\rho = 0.11\%$	$\rho = 0.07\%$

6.5 Simulation Results

The two scheduling techniques: traditional QoS algorithm (QoS) and SQ algorithm (refer to Table 4) were implemented and compared. Performance measures are presented for different number of machines and different values of λ. Each point in the graphs below is the result of 50 simulation runs. In each simulation run, a random number of tasks/subtasks for each of the three types of traffic was generated.

Simulation results are presented separately for each of the performance metrics. While section 6.3.1 shows that the SQ algorithm is working as intended by minimizing the co-allocation skew quantity, subsequent sections show how the SQ algorithm is working and what are the trade-offs, which are examined by a) determining the machine utilization in terms of the effective cycle usage, b) showing how more tasks are included with relaxed

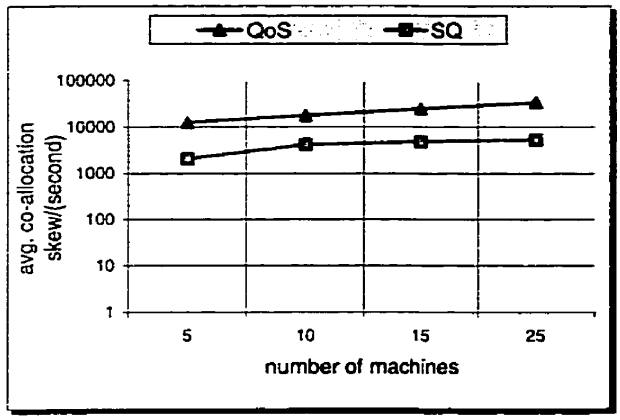
SQ admission control and what is their impact, and c) showing the ratio of the QoS conformance. The simulation results and discussion are presented below.

6.5.1 Co-allocation Skew Average

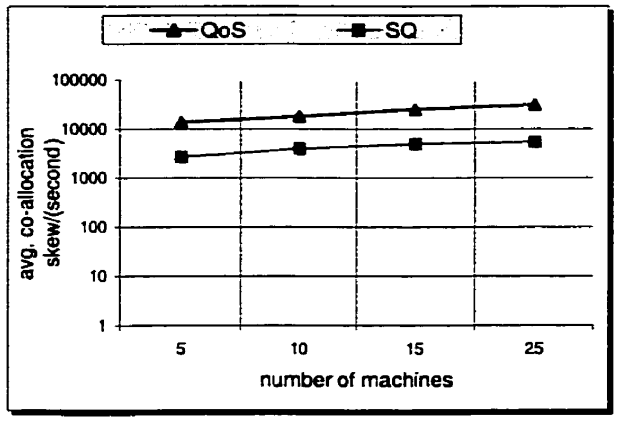
The co-allocation skew average for different number of machines and different values of λ is presented in Figure 15. This scenario is simulated for the two different algorithms presented in Table 5. It can be noted from Figure 15 that the average co-allocation skew is the highest for QoS.

Since the number of subtasks generated to form each of the GridQoS and GridBE is randomly chosen in the range of [1, # of machines], the number of subtasks will, on average, increase as the number of machines increases. Hence, the co-allocation skew for the two algorithms tends to increase with an increase in the number of machines. I chose to increase offered load with the increased number of local machines because I wanted to test SQ under both situations that cause the co-allocation problem. Particularly under the situation where the co-allocation problem is somehow an easy problem to manage (i.e. number of subtasks is small) as well as under the situation where the co-allocation problem is a much harder problem to manage (number of subtasks is much larger).

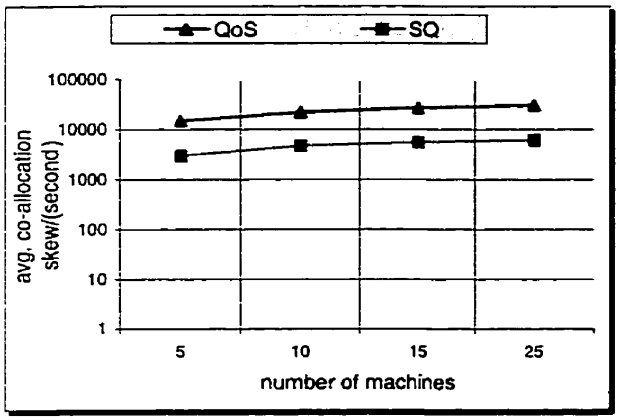
SQ has the lowest average allocation skew amongst overall. Especially compared with QoS, which uses strict admission control, SQ outperforms QoS for the different number of machines as well as for different values of λ.



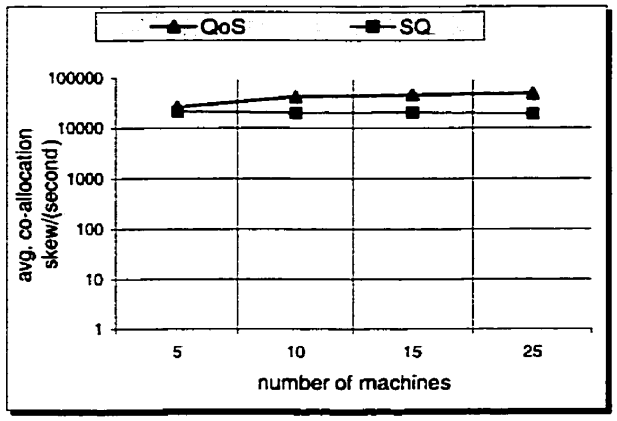
(a) Co - allocation skew variation with λ of 10 seconds.



(b) Co - allocation skew variation with λ of 100 seconds.



(c) Co - allocation skew variation with λ of 200 seconds.



(d) Co - allocation skew variation with λ of 500 seconds.

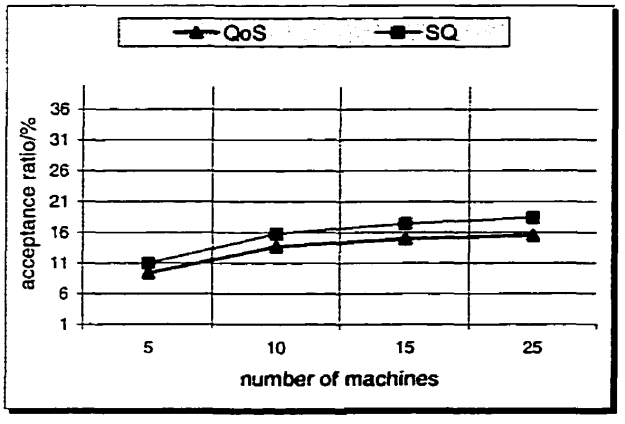
Figure 15: co-allocation skew for different number of machines.

6.5.2 Acceptance Ratio

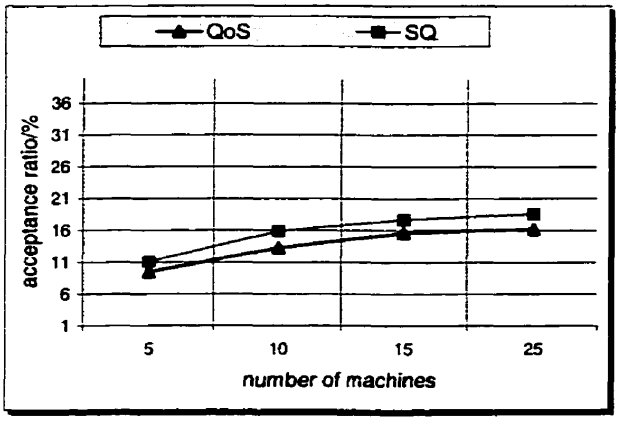
The acceptance ratio for different number of machines for the QoS and SQ algorithms with different values of λ is presented in Figure 16. Every time a QoS task/subtask is admitted to a local machine, the admission control quantity x of the local machine is decreased by the task/subtask's weight. It should be noted that admission control helps you to maintain a desired level of QoS by limiting the number of the tasks/subtasks competing for a local resource but admission control does not guarantee QoS as illustrated from the simulation study as illustrated in Figure 15. Once the quantity x is zero, no tasks/subtasks are admitted to the local machine. Once a task/subtask is finished execution, its weight is added to the quantity x, so more tasks/subtask can be admitted.

Therefore, for each algorithm, the acceptance ratio depends on the length or the lifetime of tasks/subtask assigned to the QoSQ. The length of tasks/subtasks is basically the execution time. Once the first patch of tasks/subtasks are accepted, which means that the quantity x is zero, then future tasks/subtasks can be admitted if a current task/subtask finishes execution and leaves the QoSQ. For $\lambda = 10,100,$ and 200, the Grid QoS generator is generating tasks/subtask much faster than the lifetime of theses tasks/subtasks already admitted to the QoSQ. Once the intraQueue scheduler (SFQ) for the QoSQ finishes executing a task/subtask and hence the admission control unit is in a position to admit more Grid QoS tasks/subtask, the Grid QoS generator would have finished generating tasks/subtasks. That is the reason why the results for $\lambda = 10,100,$ and 200 are somehow identical. Whereas for $\lambda = 500$ once the intraQueue scheduler (SFQ)

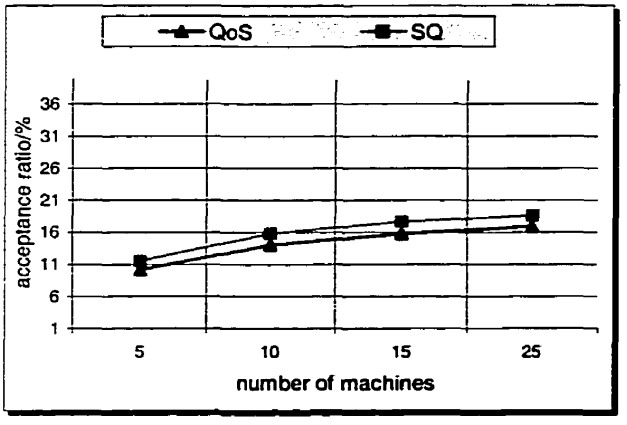
for the QoSQ finishes executing a task/subtask and hence the admission control unit is in a position to admit more Grid QoS tasks/subtask, the Grid QoS generator is still generating tasks/subtasks because the generation process is a lot slower than in the cases where $\lambda = 10,100,$ and 200. But overall, the acceptance ratio for both of the algorithms increases as the number of machines increase. Also since SQ uses relaxed admission, its ratio acceptance is overall higher than QoS.



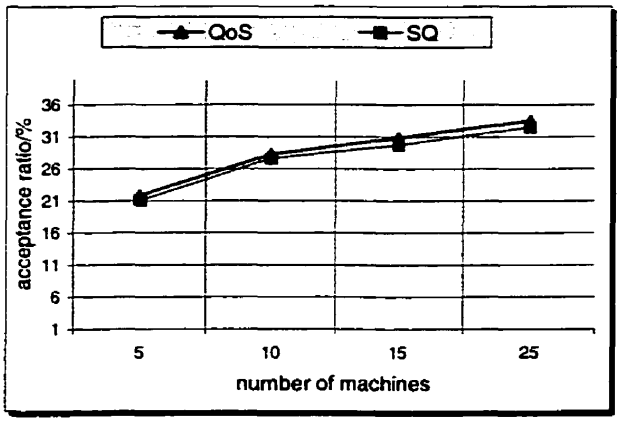
(a) Acceptance ratio with λ of 10 seconds.



(b) Acceptance ratio with λ of 100 seconds.



(c) Acceptance ratio with λ of 200 seconds.



(d) Acceptance ratio with λ of 500 seconds.

Figure 16: Variation of acceptance ratio with number of machines.

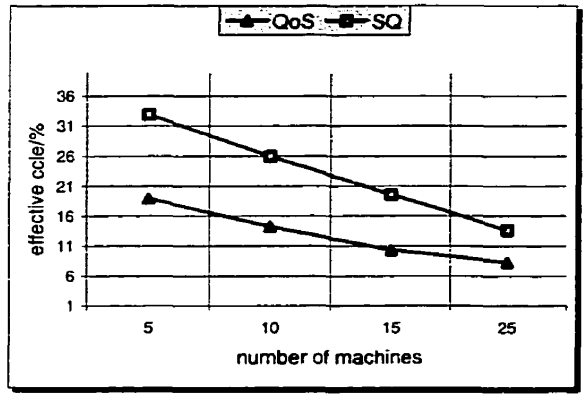
6.5.3 Effective machine usage

One of the two windows that SQ uses to synchronize hard QoS tasks is the *asynchrony* window. *Asynchrony* is one of the two QoS attributes the client provides (refer to Section 5.8). Effective machine usage is calculated as the ratio between the number of hard QoS tasks confirming to the *asynchrony* window and the total number of of hard QoS accepted. From Figure 17, it can be noticed that overall SQ has a better effective machine usage than QoS because of the corrective action taken by SQ. The fastest and the slowest subtasks of each hard QoS task are the only two subtasks from each hard QoS task that are of concern to us here.

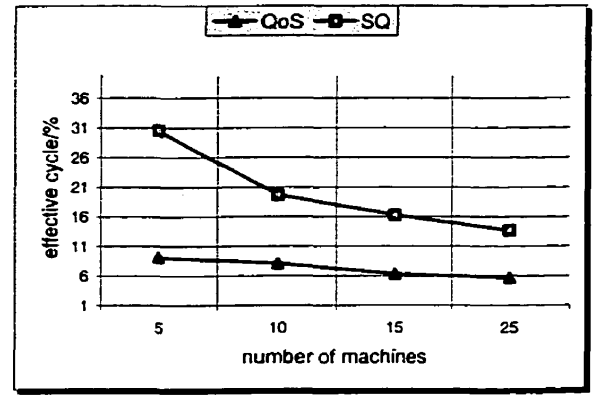
The corrective action can be slowing down or speeding up a subtask. Consider a situation where subtask s_i needs to slow down, meaning that the local machine is under loaded. In this case there are extra CPU quanta given to subtask s_i. Therefore, the extra CPU quanta can be absorbed by creating an *idle* task and that can be thought of as forcing the local machine to be idle and as a consequence wasting some of its CPU quanta. On the other hand, speeding up subtask s_i means taking back the CPU quanta needed by s_i from the *idle* task. If there is no *idle* task in the QoSQ, then the needed CPU quanta is taken from a soft QoS task/subtask if one exists.

As part of the simulation, this trade-off of forcing a local machine sometimes to have ineffective (idle) schedule cycles has been explored as shown in Figure 17. For each

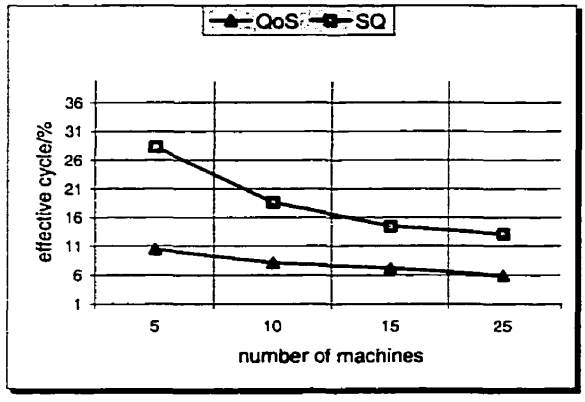
schedule cycle (monitor cycle), the asynchrony test for each hard QoS task t is performed. If it succeeds, the schedule cycle is counted as an effective cycle for task t otherwise the schedule cycle is counted as ineffective cycle for task t. As the number of the machines increase, the number of subtasks for task t increase as well and the ability to synchronize all these subtasks becomes more difficult.



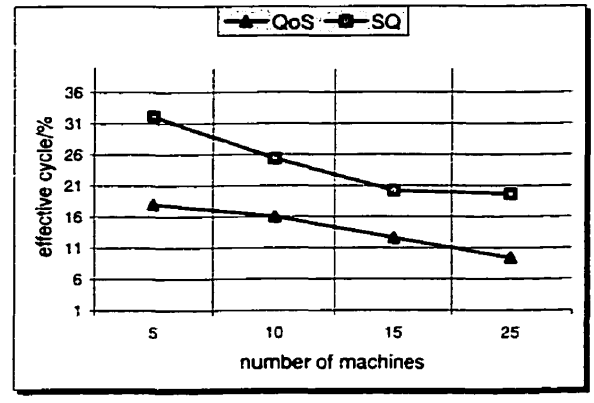
(a) Effective machine usage with λ of 10 seconds.



(b) Effective machine usage with λ of 100 seconds.



(c) Effective machine usage with λ of 200 seconds.



(d) Effective machine usage with λ of 500 seconds.

Figure 17: Effective machine usage for different number of machines.

Therefore, as presented in Figure 17, the ability of confirming subtasks to the *asynchrony* window increases as the number of machines decrease. SQ has a higher success ratio than QoS of confirming the subtasks to the *asynchrony* window, while at the same time maintaining less co-allocation skew and accepting more QoS tasks.

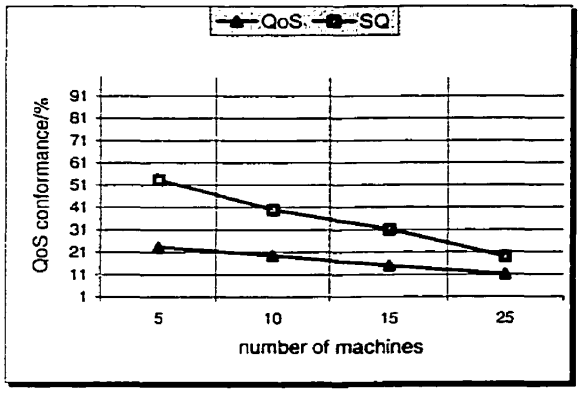
6.5.4 QoS conformance

The other window that SQ uses to synchronize hard QoS tasks is the *overall deviation* window. *Overall deviation* is one of the two QoS attributes the client provides (refer to Section 5.8). QoS conformance is calculated as the ratio between the hard QoS tasks confirming to the *overall deviation* window and the total of hard QoS accepted. From Figure 18, it can be noticed that overall SQ has a better QoS conformance than QoS because of the corrective action taken by SQ.

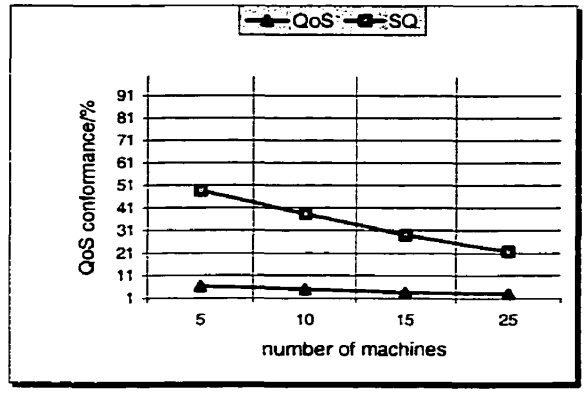
The corrective action can be slowing down or speeding up a subtask as explained in the previous subsection. From Figure 18, we observe that QoS conformance decrease with the increase of machines numbers. As the machines number increase, the number of subtasks for task t increase as well and the task of synchronization becomes more difficult due to the increase of subtasks. This phenomenon affects the co-allocation skew and the effective machine usage as well.

Therefore, as presented in Figure 18, the ability of confirming subtasks to the *overall deviation* window increases as the number of machines decrease. SQ has a higher success

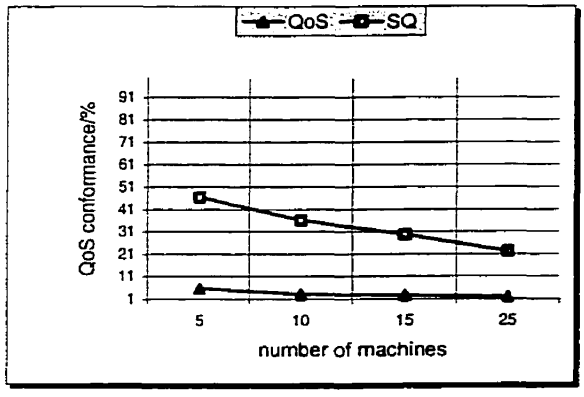
ratio than QoS of confirming the subtasks to the *overall deviation* window, while at the same time maintaining lower co-allocation skew and accepting more QoS tasks.



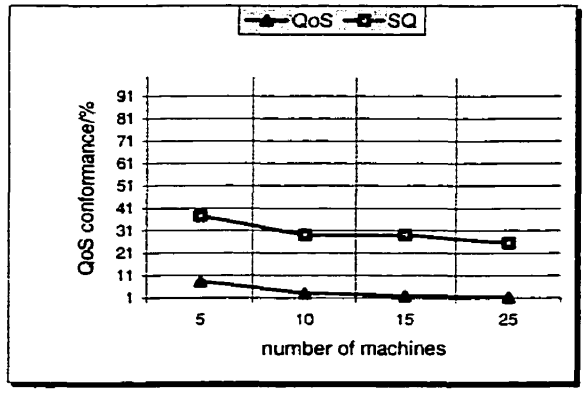
(a) QoS conformance with λ of 10 seconds.



(b) QoS conformanc e with λ of 100 seconds.

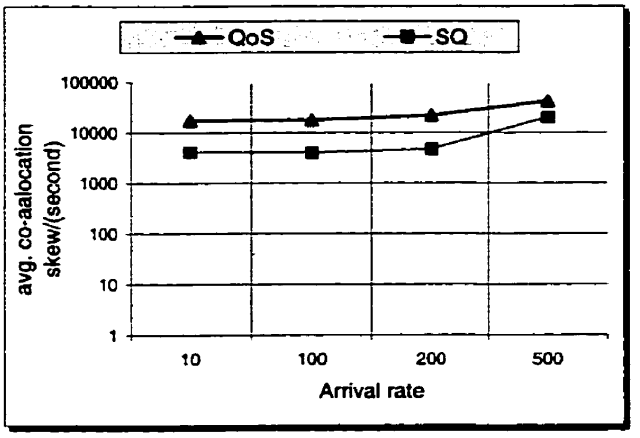


(c) QoS conformance with λ of 200 seconds.

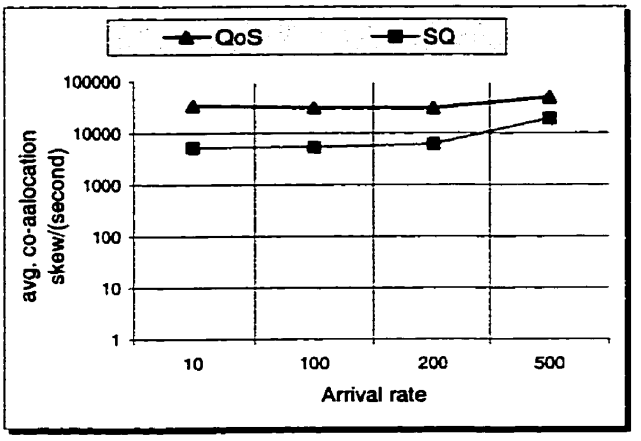


(d) QoS conformance with λ of 500 seconds.

Figure 18: QoS conformance for different number of machines.



(a) Avg. co - allocation skew for number of machines = 10.



(b) Avg. co - allocation skew for number of machines = 25.

Figure 19: Average co-allocation skew for different values of λ.

6.6 Simulation Discussion

Table 5 shows the mean value utilization analysis for the different machines with the various values of λ. From this analysis it can be concluded that the simulation performed represents underload situations and therefore are not subject to the QoS acceptance bottleneck (admission control).

As stated in Section 6.2, the goals of the simulation are to investigate and examine the co-allocation problem under different situation. Since the co-allocation problem gets more harder to manage with the increase of the number of subtasks, I chose to increase the offered load (i.e. number of subtasks) with the increased number of local machines. Also, since SQ is a feed back process, tasks/subtasks must stay in the system long enough for SQ to "kick in". Therefore, the execution time is chosen to be large enough to resemble a continuous media application so SQ, being a feed back scheme, will have enough time to "kick in" and carry its corrective action and hence be effective.

Using admission control whether strict or relaxed will allow the first batch or group of tasks/subtasks into the QoSQ and then do not accept any more tasks/subtask until a task/subtask or a group of tasks/subtasks in the QoSQ finish execution and leave the QoSQ. Once, some tasks/subtasks leave the QoSQ, more Grid QoS tasks/subtasks can be admitted. For $\lambda = 10,100,$ and 200, the Grid QoS generator is generating tasks/subtask much faster than the lifetime of theses tasks/subtasks already admitted to the QoSQ. Once the intraQueue scheduler (SFQ) for the QoSQ finishes executing a task/subtask and hence

the admission control unit is in a position to admit more Grid QoS tasks/subtask, the Grid QoS generator would have finished generating tasks/subtasks. That is the reason why the results for $\lambda = 10,100,$ and 200 are somehow identical. Whereas for $\lambda = 500$ once the intraQueue scheduler (SFQ) for the QoSQ finishes executing a task/subtask and hence the admission control unit is in a position to admit more Grid QoS tasks/subtask, the Grid QoS generator is still generating tasks/subtasks because the generation process is a lot slower than in the cases where $\lambda = 10,100,$ and 200.

The overall goal of SQ is maintained where continuous media applications, expressed in large value of , are examined and tested at situations where co-allocation skew is somewhat an easy problem to manage and at the same time I compared and investigated the situations where the co-allocation skew problem is much harder. In particular, I increased the offered load with the increased number of local machines to purely test the effectiveness of SQ and QoS schemes as the co-allocation gets harder and harder to manage (i.e. as the number of subtasks increase) as opposed to a study in which the offered load is fixed while number of machines is increased.

7 CONCLUSIONS AND FUTURE WORK

7.1 Concluding Remarks

Motivated by the successes of network computing, researchers have started examining a more generalized resource/information sharing and integration infrastructure called the Grid which is a generalized, large-scale computing and data handling virtual system that is formed by aggregating the services provided by several distributed resources.

The MEG is a concept to support the deployment of multimedia services and can potentially provide pervasive, dependable, consistent, and cost-effective access to the diverse services provided by the distributed resources and support problem solving environments that may be constructed using such resources. A user in such an environment is not tied to a specific machine but rather is a machine independent entity that exists in the Grid and can transparently carry its profile across the different platforms constituting the Grid. Some of the enabling technologies that will be supported include: (a) *quality of service* (QoS), (b) multicast, (c) streaming data, (d) co-allocation of resources, and (e) resource discovery.

This thesis addressed one of these issues, which is *co-allocation*. The co-allocation issue that is addressed in this thesis is concerned with ensuring that an application that has several subtasks would be allocated sufficient resources so that all subtasks of the application can make satisfactory progress with their execution. The co-allocation is an essential feature for several important classes of multimedia applications. One

example of these multimedia applications would be interleaved media streams where co-allocation is needed for each media stream (e.g. audio, video, color, etc) before such multimedia applications can be deployed for widespread use.

The SQ co-allocation scheme is proposed for MEGs. The contributions of this scheme are:

- a memory-oriented QoS capability: SQ is a scheme that assures the total work accomplished by each subtask s_i for each schedule cycle is accounted for. In other words, SQ is an aggregated scheme that remembers the total work accomplished by each subtask s_i in the previous schedule cycles.
- an environment-aware QoS capability: SQ is a scheme that assures the aggregated work accomplished by each subtask s_i does not fall behind the other subtasks belonging to task t. These other subtasks are running in different environments and thus it is a key point of SQ to have an environment-aware QoS capability.
- a framework for co-allocation with the ability to co-allocate heterogeneous resources in a Grid setting without the need for advance reservation.
- a framework for co-allocation with the ability to over subscribe resources and thus leading to a better utilization of the overall system than other schemes.

The algorithm and architecture for implementing SQ are presented. Simulation studies performed to evaluate SQ indicate that it outperforms admission control-based scheme by

a significant margin. The simulation studies were performed for various number of machines and inter-arrival times.

7.2 Future Work

It will be interesting to compare SQ with an *advance reservation-based* scheme where resources are reserved in advance where the client specify a start and duration of time to use the resource(s).

Also, the SQ co-allocation scheme minimizes the co-allocation skew for hard QoS tasks Future work can expand this to include Soft QoS tasks as well. . Right now if CPU quanta is needed to speed up a hard QoS subtask and there is no *idle* task in the QoSQ, then the needed CPU quanta is taken from a soft QoS task/subtask if one exists. It should be pointed out that when taking a CPU quantum or quanta from a soft QoS task/subtask, fairness is taken in consideration so that deprivations of one specific soft QoS is minimized as much as possible.

Furthermore, I chose to increase offered load with the increased number of local machines because I wanted to test SQ under both situations that cause the co-allocation problem to happen. Particularly under the situation where the co-allocation problem is somehow an easy problem to manage as well as under the situation where the co-allocation problem is a much harder problem to manage. In future work, a study in which job resource requirement is fixed while number of machines is increased can be

performed to examine if SQ is more effective than QoS in finishing the task faster while minimizing the co-allocation skew.

Therefore, in future work: (a) the SQ scheme can be compared to an *advanced reservation-based* scheme to show and compare the strengths and weaknesses of SQ; (b) the SQ scheme can be expanded to assure that the co-allocation skew of Soft QoS tasks is monitored to assure that the co-allocation skew is minimized for all the QoS tasks including hard and soft QoS tasks; and (c) a study in which job resource requirement is fixed while number of machines is increased can be performed to examine if SQ is more effective than QoS in finishing the task faster while minimizing the co-allocation skew.

References

[ArC98] A. C. Arpaci-Dusseau, D. E. Culler, and A. M. Mainwaring, "Scheduling with implicit information in distributed systems," *SIGMETRICS Conference on the Measurement and Modeling of Computer Systems,* June 1998, pp. 233-243.

[ArN99] N. Arnason, *Graduate class notes*, Department of Computer Science, University of Manitoba, ftp://ftp.cs.umanitoba.ca/pub/arnason/simjava, 1999.

[AzM00] F. A. Azzedin, and M. Maheswaran, *Design of a Quality of Service Aware API in Java,* ANRL Research Note ANRL-01-00, Department of Computer Science, University of Manitoba, 2000.

[BaB00] M. A. Baker, R. Buyya, and D. Laforenza,"The Grid: International efforts in global computing," *ACM Computing Surveys,* Oct. 2000.

[CzF98] K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke, "A resource management architecture for metacomputing systems," *4th Workshop on Job Scheduling Strategies for Parallel Processing*, Springer-Verlag LNCS 1459, 1998, pp. 62-82.

[GoG96] P. Goyal, X. Guo, and H. Vin, "A hierarchical CPU scheduler for multimedia operating systems," *Proceeding Second Symposium On Operating Systems Design and Implementation*, 1996, pp. 107-122.

[GoV96] P. Goyal, H. M. Vin, and H. Cheng, "Start time fair queuing: A scheduling algorithm for integrated services packet switching networks," *Proceeding of ACM SIGCOMM'96*, Aug. 1996, pp. 157-168.

[FeG97] D. Ferrari, A. Gupta, and G. Ventre, "Distributed advance reservation of real-time connections," *ACM/Springer-Verlag Journal on Multimedia Systems*, Vol. 5, No. 3, 1997.

[FoK98] I. Foster, C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan-Kaufmann, July 1998.

[FoK99] I. Foster, C. Kesselman, C. Lee, R. Lindell, K. Nahrstedt, and A. Roy, "A distributed resource management architecture that supports advance reservations and co-Allocation," *Proceeding of the International Workshop on Quality of Service*, 1999, pp. 27-36.

[JoG99] W. E. Johnston, D. Gannon, and B. Nitzberg, "Information power Grid implementation plan: research, development, and testbeds for high performance, widely distributed, collaborative, computing and information systems supporting science and engineering," *NASA Ames Research Center*, http://www.cs.nas.nasa.gov/IPG, 1999.

[KrM00] K. Krauter and M. Maheswaran, *Architecture for a Grid operating systems, IEEE/ACM International Workshop on Grid Computing (Grid 2000)*, Dec. 2000.

[MaK00] M. Maheswaran and K. Krauter, "A Parametric-based approach to resource discovery in Grid computing systems," *1st IEEE/ACM International Workshop on Grid Computing (Grid 2000)*, Dec. 2000.

[NiL97] J. Nieh and M. Lam, "The design, implementation and evaluation of SMART: A scheduler for multimedia applications," *Proceeding 16th Symposium on*

Operating System Principles, 1997, pp. 184-197.

[ScN99] O. Schelén, A. Nilsson, J. Norrgård, and S. Pink, "Performance of QoS agents for provisioning network resources," *Proceedings of* IFIP Seventh International Workshop on Quality of Service (IWQoS'99), London, UK, June 1999.

[Sta97] W. Stallings, *High-speed Networks, TCP/IP and ATM Design Principles*, Prentice Hall, New Jersey, 1997.

[WaW94] C. A. Waldspurger and W. E. Weihl, "Lottery scheduling: Flexible proportional-share resource management," *Proceeding First Symposium On Operating System Design and Implementation*, 1994, pp. 1-11.

[WaW95] C. A. Waldspurger and W. E. Weihl, *Stride scheduling: Deterministic proportional-share resource management*, Tech. Report. MIT, Cambridge, June 1995.

[YaD99] D. Yau, "ARC-H: Uniform CPU scheduling for heterogeneous services," *ICMCS*, Vol. 2, 1999, pp. 127-132.

[YaL96] D. Yau and S. S. Lam, "Adaptive rate controlled scheduling for multimedia applications," *ACM Multimedia Conference '96*, Nov. 1996.

[Zha91] L. Zhang, "Virtual Clock: A new traffic control algorithm for packet switching networks," Transactions on Computer Systems, Vol. 9, No. 2, 1991, pp. 101-124.