ABSTRACT:
Research in parallel simulation has been around for more than two decades. However, the number of papers reporting on its application to real world problems is limited. At the 2002 PADS conference, researchers discussed the need to go beyond synchronization and performance issues and, in particular, to demonstrate that parallel simulation could be used in real world applications outside military and network simulations. Since then, we have seen an increase in the number of papers on parallel simulation applications in areas such as operations management and the physical sciences. This paper presents a parallel discrete-event demographic simulation tool which has been developed using \( \mu \) sik parallel simulation library. A number of experiments have been conducted to evaluate the simulation performance. The result shows that good event parallelism can be achieved.

Keywords: parallel simulation, discrete-event simulation, simulation tool, population dynamics, demography, performance evaluation

1. INTRODUCTION
In the past two decades, Parallel Discrete-Event Simulation (PDES) research has been dominated by the development of better synchronization algorithms and their performance evaluations. Researchers at the 2002 Workshop on Parallel and Distributed Simulation (PADS) in Washington, D.C. discussed “life beyond synchronization”. It was felt that it was high time for researchers to promote the use of parallel simulation in wider areas of application. Traditionally, parallel simulation has been applied in military and network simulations. Since then, we have seen an increase in the number of papers reporting on parallel simulation applications outside traditional areas. Tang et al. (2005) conducted an initial study in applying parallel simulation to a plasma physics application. In the realm of biological science, Lobb et al. (2005) applied parallel simulation to a neuron model. Lan and Pidd (2005) applied parallel simulation to simulate a quasi-continuous manufacturing process. Yoginath and Perumalla (2008) applied parallel simulation to a traffic simulation model. Bauer et al. (2009) conducted an experiment to evaluate the scalability of their parallel simulation tool using a Transmission Line Matrix model (for electromagnetic wave propagation). Park and Fujimoto (2009) evaluated their Master/Worker parallel simulation tool using a particle physics model. The situation is encouraging, although it is still far from ideal, where parallel simulation is widely used across many different application areas including the social sciences. Such diversity of application motivates us to study the feasibility of using parallel simulation in demography. Demography is often used as the basis for government policies in areas such as education, healthcare, social welfare and taxation. For example, a fall in the number of births may lead to school closures. Similarly, a rise in the number of elderly people may lead to an increase in the demand for healthcare services related to the elderly.

In the past few years, supported by the advances in computer technology and the availability of data at a micro level (individuals), the use of micro-level simulation models in population projection has become more widespread. The main advantage of this approach is that individual-specific explanatory variables can be included in the model. For example, we may include factors such as age, education level, salary group and ethnicity to model the number of children that an individual female will have. The projected population often used as an input to policy models. Traditionally, policy models allocate people in every population group (usually based on age and gender) to a set of states (for example, in full-time employment, in full-time education, unemployed and retired). As computer technology advances and the data at a micro-level have become available, the multi-state simulation model at the level of individuals has gained popularity for policy analysis. In this model, the state transition of each individual is simulated (such as the transition from being in full-time education to full-time employment). The
multi-state model has now become the standard methodology in demography (Willekens, 2005). Complex policy models that include biological factors (such as health-related factors), cognitive factors (such as learning) and social factors (such as social network) may require a significant amount of computing power. As shown later, even if the demographic model alone requires a significant amount of computing power, let alone if a complex policy model is added on top of it. Parallel simulation may offer an alternative solution.

This paper presents some improvements that have been made to the parallel discrete-event demographic simulation tool reported in Onggo (2008). The improvements include the change in the Logical Process (LP) design, the use of LP pooling, and a facility which enables modellers to specify key demographic components such as fertility, mortality and migrations. The rest of the paper is organized as follows. Section 2 provides an overview of various demographic components that are supported in the simulation software. This is followed by an explanation on the design and implementation of the simulation software. Section 4 presents the performance evaluation results of the simulation software. Finally, the concluding remarks and suggestions for further work are given in Section 5.

2. DEMOGRAPHIC MODEL

Demography is the study of human population in relation to changes brought about by the interplay of births, deaths and migration (Pressat, 1985). Population projection is one of the main applications of demography. Traditionally, there are two methods commonly used in population projections: mathematical model and cohort component (Hinde, 1998). In the first method, demographers use a set of mathematical equations to project the size of a future population. The cohort component method is used to project the structure of a population, i.e., the size of different groups within the population (such as different age groups). This is an iterative method in which each iteration projects the structure of a population in the following year using a set of flow models. Hence, the latter method is able to produce a more detailed projection. For many purposes, the population structure is more useful than the total population size. The use of simulation for population projection has increased in recent years. As stated previously, this is because we may include individual-specific explanatory variables in the model. To take one example, Liddle (2002) developed a simulation model to assess the effect of demographic dynamics on sustainable development at three levels: economic, social and environmental. Apart from their application in population projection, demographic models are often used as the basis for policy modelling and analysis. To take two cases, Walker et al. (2000) used simulation to analyze the effects of demographic changes on government expenditure on pharmaceutical benefits in Australia. Similarly, Bonnet and Mahieu (2000) analyzed pension policy in France using simulation. Their simulation model comprised three main components: demographic, labour market and income.

2.1 Components of Demographic Model

The main core of a demographic simulation model is the life course of individuals. In general, a demographic simulation model is formed by a number of demographic components such as birth (fertility), change in economic status and marital status, migration and death (mortality). Hinde (1998) provides a good explanation of various demographic components and their modelling methodology. A short summary is given below.

Common models for the birth component are related to the fertility model that describes the propensity of the women in a population to bear children. The models include age-specific fertility, parity-specific fertility, birth spacing and their combinations. Age-specific fertility uses age to determine the probability of having a child. Parity-specific fertility takes into account the number of children a female individual has already had. The birth spacing model focuses more on the time between each birth. It is rather complicated to model because it includes factors such as birth control, abstinence period, economic status, etc.

The ability to track an individual’s economic status and marital status is included in the model because there is a plan to build a policy model on top of it (for public policy planning and analysis in such areas as taxation, pensions and benefits). This ability is essential because many policies are linked to each status. Most models use state-transition diagrams to represent the possible changes in status. In the simplest case, the transitions are made based on a series of simple random samplings. More complex models will include explanatory variables such as an individual’s characteristics, the characteristics of the individual’s family and external socio-economic factors. In demographic simulation, the marriage model is rather special. Two marriage models are commonly used: open and closed. An open marriage model is simpler because it simply changes the marital status of the individual in the model. A closed marriage model, which is used in our simulation tool, is more complicated. In this
model, the simulation can only schedule the start of a ‘partner search’ rather than a marriage per se. If a suitable individual is found from the list of prospective partners, then the marriage event will be scheduled for both individuals. Otherwise, the individual will be added to the list of prospective partners for a specific duration. The same model applies for cohabiting individuals and civil partnerships.

The effect of migration has become more significant in recent years. Unlike births and deaths, an individual may migrate (either domestically or internationally) more than once. A simple model uses a simple random sampling to decide whether an individual is going to migrate and to determine the new place. A more complex model employs a combination of individual-specific factors, family-specific factors, region-specific factors and other external factors to explain the individual’s decision to migrate to a new place.

The most commonly used method in mortality analysis is the life table (and its variants). The life table summarizes the variation of mortality by age and gender. Survival analysis is another commonly used method where a distribution function of lifetime was used to sample an individual’s lifetime at birth.

2.2 Tools for Demographic Simulation

One of the commonly used paradigms in demographic simulation is microsimulation. The initial work in microsimulation goes back to the work of Orcutt (1957). In this simulation paradigm, modellers have to specify a random sampling process for each individual at each simulation time point, to determine the state of each individual at the next simulation time point. At one extreme, the sampling process requires a simple random sampling. At another extreme, it may require a complex regression model. However, most microsimulation tools have been built for certain public policies. Examples include LABORSim for policies related to labour supply in Italy (Leombruni and Richiardi, 2006) and Pensim2 for the British pension system (O’Donoghue and Redway, 2009). SOCSIM (Hammel et al., 1990) is one among the few generic microsimulation tools for demography. Dahlen (2009) develops of an open-source microsimulation tool called LaMPsim that has been designed for labour market policy in Sweden. Being an open source, it might be possible to extend the software for a more generic application. System dynamics is another commonly used paradigm in developing demographic simulation models. Important works in this area include the World Dynamics (Jay Forrester, 1971) and World3 population model (Meadows et al. 1972, 2004). Unlike microsimulation and system dynamics, most demography-related discrete-event simulation tools are built for health-care related cases (Eubank, 2002; Rauner et al., 2005; Rao and Chernyakhovsky, 2008). Some of the tools support parallel simulation execution (Eubank, 2002; Rao and Chernyakhovsky, 2008). The tool described in this paper focuses on the economic and social transitions of population. Hence it is different from Eubank (2002) and Rao and Chernyakhovsky (2008) that focus on the spread of epidemic.

3. DESIGN AND IMPLEMENTATION

The demographic simulation software was implemented using μsik parallel simulation library (Perumalla, 2005). This library supports multiple synchronization algorithms such as: lookahead-based conservative protocol and rollback-based optimistic protocol (state-saving and reverse-computation). It has been reported to be scalable to run a synthetic benchmark called PHOLD on a large number of processors (Perumalla, 2007). This library adopts the process interaction worldview; hence a simulation model is formed by a set of interacting (logical) processes. Logical processes (LPs) communicate through events. Multiple LPs are mapped onto a physical process (PP) that is run on top of a processing element (PE). A machine can have more than one PE (e.g., in multi-core architecture). It should be noted that in μsik documentation, PP is often referred to as federate. To avoid confusion with the federate in High Level Architecture, in this paper the term PP is used instead of federate.

To implement a simulation model in μsik parallel simulation library, we must specify three main components: a physical process (must inherit from class Simulator), a set of logical processes (each must inherit from one of these classes: NormalSimProcess, PeriodicSimProcess, or ThreadedSimProcess), and a set of events (each must inherit from class SimEvent). A detailed explanation on the structure of a simulation model written in μsik can be found in Perumalla (2005). The following sub-sections explain how the three main components are implemented for the demographic simulation software.
3.1 Physical Process

The physical process, implemented as class PopulationSimulator, is defined as a subclass of class Simulator. The main tasks of this class are to: establish the simulation parameters, generate initial population, manage LPs and generate simulation reports. Figure 1 shows that the simulator will initialize a number PPs, each of which will run on a PE. This is followed by the initialization of two types of LPs: Region and FamilyUnit (FU). Each LP Region represents an administrative area where a number of families live. Each LP FamilyUnit represents a family in the population. In this paper, we use a strict definition of family which will be discussed later. Hence, for each PP, there will only be one LP Region and a number of LPs for FamilyUnit. Communication between two LPs occurs when an LP (in this case a family) from a region in one PP wants to migrate to another region on a different PP.

3.2 Logical Processes

One of the key design decisions concerns the types of LPs that are going to be used in the model. In the earlier work (Onggo, 2008) each individual was represented as an LP. This turns out to be problematic. The reason is that public policies may apply to individuals as well as groups of related individuals, such as households and single parents. For example, the UK Department for Work and Pensions and HM Revenue & Customs manage a number of public funds that may apply to individuals (including jobseeker’s allowance and incapacity benefit) or groups of related individuals (which could include child benefit and housing benefit). Therefore, it is important that the model recognizes different types of ‘policy unit’. Policy unit is often referred to as ‘family unit’. A family unit is formed by either a single independent individual or two independent individuals living together (as married, in civil-partnership, or in cohabitation) and any dependent individuals (children). Hence, in this definition, a

![Figure 1 LP structure and communications](image-url)
family unit may represent an independent individual, a single parent, a childless couple or a nuclear family. For completeness, the definition is extended to include orphans, that is, a family unit of dependent children without any parents.

The decision to represent a family unit as an LP has another advantage. When there is a change in the marital status that affects couples (such as from married to divorced or from married to widowed), only one message needs to be sent to the affected couple. In the earlier work, two messages had to be sent, one for each affected individual. Hence, it reduces the number of sent messages in the simulation.

A family unit may receive events which are related to five demographic components that may change the system states. Modellers need to specify models for five demographic components: fertility, a change in economic status, a change in marital status, migration and mortality.

The fertility component determines whether a female individual will give birth, based on the characteristic of the female individual and the current calendar time. The model returns the time when the baby is due. Similarly, the characteristic of an individual and the current calendar time determines a new economic status that will be effective at time from now.

A new marital status that will be effective at time from now depends on the characteristics of the individual (or individuals for a couple) and the current calendar time. If the new status is either married or cohabitating, modellers need to define the criteria that will be used to match the individual to another individual from the list of prospective partners (i.e. closed marriage model). If a suitable partner is found, then a ‘family formation’ event will be scheduled for both individuals at time. Otherwise, the individual will be added to the list for a fixed duration. If a partner still cannot be found at the end of the duration, an event will be sent to remove the individual from the list. Modellers need to specify model that is used to determine whether a family unit is going to migrate. If the destination is in another country (emigration), the family unit will simply be removed from the simulation. Finally, in the mortality component, modellers need to model the time when an individual will die based on the characteristics of the individual. Commonly used methods, such as life table and survival function can be used for the mortality component.

Finally, the competing risk model (Hosmer et al., 2008, Chapter 9) where every family will have exactly one future event is used. This approach will sample time-to-event for a number of competing events such as death, giving birth, change in marital status and change in economic status. The event with the shortest time-to-event will be chosen and executed. This process is repeated whenever a life event occurs (except for death and emigration).

As in the previous work, an administrative area where a number of families live is implemented as an LP that is called Region. This LP will handle domestic migrations, immigration, changes in simulation parameters and periodic reports. At the beginning of the simulation, an identification number is assigned automatically to each region. In the future version of this tool, this identification number will be used to model regions with different characteristics.

A region may receive four types of events. When a family unit is going to migrate to a region, the family unit will send a FUResult event to the destination region. The destination region will prepare an empty LP FamilyUnit and send the LP’s identification number to the migrating family so that the members of the family can be transferred to the new LP. The immigration event is used to simulate family units entering the country every month (in batches). The tool allows modellers to implement different models for immigration policies, such as the number and demographic characteristics of the migrants. Event ParamsChange allows modellers to specify periodical changes in simulation parameters such as life table and fertility rate. Finally, the report event is called periodically. Each time this event is received, a report on the population structure (by gender, age group, marital status and economic status) is generated.

3.3 Simulation Outputs

A validation of the individual-based population model in itself is a complex and important topic. This is partly due to the fact that model parameterisation is challenging and time consuming. Many periodical censuses and surveys change their methodology from time to time, which makes data comparison difficult; for example, the definition of still birth in the UK was changed on 1st October 1992. What the author has done at this stage is to verify whether the output of the simulation is sensible. The author has conducted a few experiments and plots their results for analysis. Two of the results are shown in Figure 2.

The figure on the top is the population pyramid of the initial population which is based on the UK
Family Resource Survey data in 2008/09. A simple combination of age-specific fertility, parity-specific fertility and birth spacing are used for the fertility model (representing a population with an average fertility rate of 1.3). The UK life table based on the 2005-2008 data is used for the mortality model. A simple random sampling method is used for domestic migrations and changes in marital status and economic status. Finally, there are no international migrations (representing a closed population). The bottom figure shows the population pyramid ten years later. This setting provides an example of how a low birth rate affects the demography of a closed population in 10 years.

Figure 2 Population pyramids (top: initial population, bottom: 50 years later)

4. PERFORMANCE ANALYSIS

A number of experiments have been conducted to understand the effect of population size and migration activities on performance and the scalability of the tool. All experiments were run using µsk settings that gave a roll-back based optimistic parallel simulation execution with a state-saving mechanism and a time window of 12 months (to limit how far an LP can advance ahead of others). Fujimoto (2000, Chapters 4 and 5) provides a good overview of various techniques in optimistic parallel simulation. The model uses a continuous time where future events can happen almost immediately. The lookahead is relatively small that makes a conservative protocol less efficient. For this reason, the optimistic protocol is used. The program was compiled using gcc version 4.1.2 with the optimization flag O3 turned on and openMPI version 1.3.1 was used. All experiments were run on a cluster of PCs connected via a dedicated gigabit Ethernet switch. Each node has two quad-core 2.3GHz AMD Barcelona CPUs and 16GB of memory. All experiment results presented in this section are based on the average of five replications.

4.1 Effect of population size and migrations on execution time

Perumalla (2005, 2007) has carried out a number of experiments to evaluate the performance of µsk simulation library. The following experiments have been designed to understand the effect of population size and the level of migrations on the overall simulation performance.

Firstly, the effect of the number of family units on computation time is measured by disabling the migrations. The simulation for a period of 50 years was run with different initial population sizes of 160,000, 320,000 and 640,000 family units. The number of individuals is approximately twice the number of family units. Since the average fertility rate is set around 1.3 with no immigration, the numbers of family units and individuals at the end of the simulation are approximately 20% less than their initial size. Figure 3 (top) shows how the increase in number of family units increases computation time. The experiment on the sequential execution shows that without any migrations in the population, the total overhead is relatively constant (around 8 seconds) and is negligible for big problem sizes. It also shows that an increase the number of family units has a significant effect on the increase in computation time.

Secondly, the effect of migrations on execution time is measured. The simulation was started with 640,000 family units and was run for a period of 50 years on one computer node (i.e. up to eight processors). The probability of migrations was varied between 0% and 60%. The results are shown in Figure 3 (bottom). When the probability
of migrations is 20% or lower, the execution time decreases as we increase the number of processors. The reduction in the execution time becomes less significant as the number of processors increases. This is because the reduction in the computation cost becomes less significant and at the same time the communication costs becomes more expensive as the number of processors increases.

Figure 3: Effect of population size and degree of migration on execution time

When the probability of migrations is 40% or higher, it follows the same pattern except when two processors are used. The same pattern was also observed when the experiments were conducted on two different PC clusters with different configurations of hardware and operating systems. An initial observation indicates that the population size has an exponential effect on the computation costs of methods that handle migrations. When two processors are used, the costs for non-migration-related computations are reduced significantly (as shown in Figure 3 left). However, it is followed by a massive increase in the costs for migration-related computations. As a result, the overall reduction in computation costs is rather moderate. At the same time, migrations incur a significant amount of communication costs. Further investigation is needed to understand this behaviour.

4.2 Effect of population size and migrations on scalability

The objective of this experiment is to study the effect of varying the number of processors and the inter-processor communications on the event parallelism for a fixed population size (strong scaling) and to study the event parallelism as the number of processors and the population size are increased in proportion (weak scaling).

Figure 4 (top) shows the event parallelism when the simulation is run for 50 years with 640,000 family units. The result shows that, for the same problem size, an increase in the number of processors increases computing power, but at the same time more synchronization overheads are required. This explains the diminishing performance gain as we increase the number of processors. For the same number of processors, an increase in the proportion of family units who are going to migrate, from 20% to 60%, reduces the exploited event parallelism. This is due to the increase in the number of inter-processor communications. This also explains why the performance gain diminishes faster for the higher proportion of migrations.

For the weak scaling (Figure 4, bottom), the simulation was run with 10,000 family units on one processor and increased the number of family units and processors proportionally up to 640,000 family units running on 64 processors. The
simulation run length was fixed at 50 years. An increase in the number of processors increases the computing power, while an increase in the number of family units increases the number of migrations (hence, inter-processor communications). Thus, in this experiment, we want to see whether the increase in computing power can cope with the increase in the communication workload. The result shows that at 10,000 family units per processor, the increase in computing power can cope very well with the increase in inter-processor communications up to eight processors because the inter-processor communication in one node is very fast. As we increase the number of processors and family units further (beyond the node boundary), the exploited event parallelism decreases. This is due to the higher cost of inter-node communication in the cluster.

5. CONCLUSIONS & FURTHER WORK

We have presented a novel application of parallel discrete-event simulation in demography. This paper has discussed a tool that will help modellers to specify key demographic components and run a large-scale simulation on parallel computers. We have explained a number of improvements to our earlier work. First, the tool has provided a set of virtual functions for key demographic components that can be overridden by modellers to specify their models. However, at this stage, the specification has to be coded manually. In the future, this will be replaced with a graphical user interface where model developers can specify the components of the models visually. Second, the design of the Logical Processes (LPs) has been changed from representing individuals to families. This reduces the complexity of the program and improves the simulation performance. Finally, we have conducted a set of experiments to evaluate the performance of the tool. The result looks promising. For example, the sequential simulation of 640,000 families over 50 years with a reasonable level of migrations (20%) takes just over 2 hours, but on 16 processors it takes less than 4 minutes. There is room for improvement since the scalability of the tool decreases significantly when we use processors in different nodes. Further work is needed to understand and to optimize the tool performance on an execution platform with a varying level of communication costs. A more thorough model validation needs to be done. Researchers in statistical demography have proposed a number of methods that could be used to validate demographic models (for example, see Bohk et al. 2009; Zinn et al. 2009).

ACKNOWLEDGEMENT

This research is funded by the Lancaster University small grant SGS/14/17 (2008).

REFERENCES


AUTHOR BIOGRAPHY

BHAKTI S. S. ONGGO is a lecturer at the Department of Management Science at the Lancaster University Management School. He received his MSc in Management Science from the Lancaster University and completed his PhD in Computer Science from the National University of Singapore. His research interests are in the areas of simulation methodology, simulation technology and simulation applications. His current research includes the application of parallel simulation in public policy analysis and simulation conceptual model representation.