ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ С ПРОГРАММИРУЕМОЙ СТРУКТУРОЙ
(Вычислительные системы)
1982 год Выпуск 94

УДК 681.324

ОПЫТ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ОРГАНИЗАЦИИ МЕЖМАШИННЫХ ВЗАИМОДЕЙСТВИЙ В СИСТЕМАХ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ИНФОРМАЦИИ

А.В.Арандоренко, С.И.Гордеев, В.В.Склянкин, И.П.Шилкин

Современные требования, предъявляемые к АСУ ТП сложных производств, таковы, что они не могут быть выполнены при использовании отдельных нерезервированных управляющих машин. Так, например, для управления ответственными физическими установками среднее время наработки на отказ управляющего комплекса технических средств должно составлять десятки тысяч часов. Отсюда вытекает необходимость создания многомашинного резервированного вычислительного комплекса.

Перед научной группой Московского инженерно-физического института была поставлена задача разработки программного обеспечения для такого комплекса с единой информационной базой емкостью несколько Мбайт с резервированием ЭВМ и автоматическим замещением отказавшего оборудования. В ходе исследований были рассмотрены вопросы обеспечения как аппаратной, так и программной надежности вычислительного комплекса.

Указанная задача решалась в два этапа. На первом этапе было разработано программное обеспечение двухмашинного вычислительного комплекса с общей внешней памятью на магнитных дисках. На втором - решалась проблема разработки программного обеспечения для сети ЭВМ. При этом на первый план выдвигалась необходимость создания программно-доступной линии межмашинной связи, поскольку особенности ЭВМ М-6000 не позволяют без разработки дополнительной аппаратуры создать общую внешнюю память на магнитных дисках более, чем для двух ЭВМ.

Итогом первого этапа работ явилось создание и внедрение в промышленную эксплуатацию на ряде предприятий операционной системы (ОС) МАРС (не следует смешивать с Модульной Асинхронной Развиваемой Системой, разработанной в ВЦ СО АН СССР). Система МАРС предназначена для использования в двухмашинном вычислительном комплексе, построенном на базе ЭВМ М-6000. Надежность функционирования комплекса обеспечивается путем горячего резервирования основного оборудования комплекса (процессора, оперативной памяти, накопителей на магнитных дисках, пультов оператора).

Общие принципы организации межмашинного обмена в ОС МАРС

Обмен информацией между ЭВМ двухмашинного комплекса в ОС МАРС может осуществляться двумя способами:

- через общую для двух ЭВМ внешнюю память на магнитных дисках;
- через линии межмашинной связи.

Обмен через общую внешнюю память доступен для всех функциональных программ, работающих под контролем ОС МАРС, и осуществляется за счет создания общих дисковых массивов. При необходимости можно регламентировать доступ функциональных программ к общим массивам, используя систему блокировок и другие защитные механизмы.

Обмен через линии межмашинной связи доступен т о л ь к о самой операционной системе и используется для организации взаимодействия ЭВМ при работе с общим оборудованием и ряда других целей. Для обеспечения эффективной работы в ОС МАРС реализована обработка привилегированных прерываний от линий связи.

Двухмашинный вычислительный комплекс может функционировать в следующих двух основных режимах:

- в симплексном режиме, когда работает лишь одна ЭВМ;
- в дуплексном режиме, когда работают две ЭВМ комплекса.

При работе в дуплексном режиме ОС использует линии связи для передачи сигналов времени, системных таблиц, требований на блокировку и разблокировку дисковых массивов и устройств и т.п.

Обработка принятой информации осуществляется по принципу "почтового ящика". При этом завершение передачи отмечается установкой некоторых флагов. Если при работе в дуплексном режиме произошел отказ одной из ЭВМ, то вторая ЭВМ переходит в работу в симплексном режиме, реорганизуя соответствующим образом вычислительный процесс.

Опыт показывает, что после внедрения ОС в эксплуатацию потребители по мере освоения ими возможностей системы увеличивают количество функциональных задач до тех пор, пока дальнейшее увеличение вычислительной нагрузки становится невозможным без ввода в действие дополнительных ЭВМ. Появление дополнительных ЭВМ поставило перед разработчиками новые проблемы. Возникла, в частности, необходимость в обмене данными между функциональными программами, выполняющимися в разных ЭВМ.

Для решения этой задачи было решено соединить ЭВМ, участвующие в вычислительном процессе, линиями межмашинной связи. Для близких расстояний (до 50 м) линия связи может быть реализована на дуплексных регистрах, а на больших (до I км) - на модулях быстрой передачи данных.

Как известно [I], одним из возможных полных наборов системных операций коллективного взаимодействия вычислителей являются следующие операции:

I) н а с т р о й к а, связанная с заданием схемы связей между вычислителями в соответствии со структурной схемой информационных связей между задачами (и частями задачи);

2) о б м е н н ы е в з а и м о д е й с т в и я между вычислителями;

3) у п р а в л е н и е параллельным процессом вычислений.

При этом наиболее важную роль играют операции обменных взаимодействий, которые и рассматриваются ниже. К программному обеспечению программно-доступной линии связи помимо общих требований по реализации полного набора системных операций были предъявлены также дополнительные требования:

- инвариантность вычислителей, участвующих в обмене, к типу ОС;

- минимальные изменения в ОС.

Для удовлетворения этих требований было принято решение реализовать программное обеспечение программно-доступной линии связи в виде драйвера, выполняющего системные операции обмена и управления и дать типовой алгоритм обмена.

Поскольку объемы массивов передаваемой информации могут быть достаточно большими (несколько тысяч слов), а размеры оперативной памяти ЭВМ М-6000 невелики (до 32 Кслов), то организация обмена по принципу "почтового ящика" была признана неподходящей. Обмен информации осуществляется сразу из тела программы в тело програм-

мы, минуя промежуточный буфер. Так как процессы в вычислителях протекают асинхронно, то первое с требованием обмена может обратиться как передающая, так и принимающая ЭВМ. Развязывание коллизий осуществляется с использованием блокировок и тайм-аута.

Для управления ходом вычислений в противоположной ЭВМ драйвер реализует функцию планирования программы в противоположной ЭВМ по имени программы с передачей управляющих параметров.

Типовой алгоритм обмена информацией заключается в следующем. Программа-инициатор планирует на соседней ЭВМ программу-партнера, с которой она будет вести обмен данными и, если планирование осуществилось успешно, вступает в обмен. Если требуется защититься от вмешательства со стороны других программ, то перед началом операций программа-инициатор блокирует линию связи с обеих сторон, позволяя обмениваться данными только программе-инициатору и программе-партнеру, а после окончания операций разблокирует ее.

Опыт разработки и эксплуатации показал, что на технических средствах типа ЭВМ М-6000 можно строить высоконадежные комплексы и организовывать эффективное взаимодействие как на уровне общей памяти на магнитных дисках, так и на уровне программно-доступной линии связи. В качестве примера укажем, что время наработки вычислительного комплекса на отказ при работе в симплексном режиме составляет около 100 часов, при работе в дуплексе - 6000-8000 часов.

Выбор оптимальной длины сообщения, передаваемого по линии межмашинной связи

Одним из вопросов, которые приходится решать при проектировании распределенных систем обработки информации, является вопрос борьбы с ошибками оборудования. Как известно [2], ошибки в каналах передачи данных возникают гораздо чаще, нежели в других элементах комплекса. Поэтому имеет смысл учесть влияние именно этих ошибок на качество функционирования распределенных систем.

Наиболее распространенным способом обнаружения ошибок при обмене является формирование контрольных сумм на передающей и приемной стороне и их последующее сравнение. В случае несовпадения используется обычно повторная передача искаженного сообщения. При этом накладные расходы (время, затрачиваемое на подготовку к повторной передаче) существенно снижают скорость обмена. Поэтому следует вести передачу длинными блоками. Однако, если блок данных будет слишком длинным, время, используемое для повторной передачи,

становится недопустимо большим. Следовательно, существует длина блока, для которой потери времени на повторную передачу минимальны.

Рассмотрим канал связи с автоматическим повторением искаженных блоков данных.

Среднее время на передачу блоков в канале составит

$$t_{ср} = t_з + \frac{N}{S} + \left(t_п + \frac{N}{S}\right)(P + P^2 + P^3 + \ldots), \qquad (1)$$

где N - число слов в передаваемом блоке (вместе с контрольной суммой); $t_з$ - время, затрачиваемое на подготовку блока к передаче (время загрузки блока); $t_п$ - время, затрачиваемое на подготовку блока к повторной передаче после сбоя; S - скорость передачи по линии связи; P - вероятность наличия ошибки при передаче блока.

Последнее слагаемое в правой части (I) представляет собой математическое ожидание времени, затраченного на повторение передачи при возникновении ошибок. Поскольку $P + P^2 + P^3 + \ldots = \frac{P}{1-P}$ при $P < 1$, то

$$t_{ср} = t_з + \frac{N}{S} + (t_п + \frac{N}{S})\frac{P}{1-P}.$$

Эффективная скорость передачи данных по линии связи составит

$$S_э = \frac{N}{t_{ср}} = \frac{N}{t_з + \frac{N}{S} + (t_п + \frac{N}{S})\frac{P}{1-P}}.$$

Положив для простоты $t_з = t_п$ (часто встречающийся на практике случай), будем иметь

$$S_э = \frac{SN(1-P)}{N+St_з}.$$

Если вероятность ошибки при передаче одного символа обозначить через P_c, то вероятность появления ошибки при передаче блока из N символов будет равна $P = 1 - (1-P_c)^N$. С учетом этого эффективная скорость передачи данных окажется равной

$$S_э = \frac{SN(1-P_c)^N}{N + St_з}.$$

Поиск длины блока, максимизирующей значение $S_э$, аналитическими методами затруднителен. Поэтому при решении задачи использовались численные методы. Оптимальная длина блока была определена для трех устройств обмена из номенклатуры АСВТ-М и СМ ЭВМ:

- модуля внутрисистемной связи А723-5;
- модуля быстрой передачи данных А723-1;
- дуплексного регистра А491-3М.

Результаты представлены в следующей таблице

Наименование устройства	Скорость передачи, слов/сек	Вероятность ошибки	Время перезагрузки, мс	Оптимальная длина блока, слов
Модуль внутрисистемной связи А723-5	$6,1 \cdot 10^5$ $2,5 \cdot 10^5$ $4,5 \cdot 10^4$	$5 \cdot 10^{-4}$	2	1000 600 500
Модуль быстрой передачи данных А723-1	$6,1 \cdot 10^4$	$5 \cdot 10^{-4}$	2	520
Дуплексный регистр А491-3М	10^4	$5 \cdot 10^{-4}$	2	190

Методы оценки программной надежности и их применения

При проектировании программного обеспечения обмена в многомашинных вычислительных комплексах необходимо учитывать характеристики надежности не только аппаратных средств, но и разрабатываемых программ организации обмена. Отметим, что ориентировочное определение надежностных характеристик программного обеспечения требуется уже к моменту его передачи в опытную эксплуатацию. Существуют несколько моделей прогнозирования количества скрытых ошибок, оставшихся в программном обеспечении после окончания отладочного периода. Все их можно свести к двум следующим типам:

1) структурные модели, использующие для оценки программной надежности понятие сложности структуры программного обеспечения [4];

2) статистические модели, в которых заключение о количестве скрытых ошибок принимается на основании результатов статистиче-

ского анализа процедуры отладки. Большинство этих моделей носит вероятностный характер [3].

Применение статистических моделей возможно лишь при наличии обширной статистики, отражающей процесс накопления ошибок в период отладки. Набор такого рода статистики и "ручная" привязка обнаруженных ошибок ко времени затруднительны (особенно в условиях напряженной отладки). Поэтому при использовании статистических методов прогнозирования надежности программного обеспечения следует ориентироваться на автоматические (программные) средства набора отладочной статистики.

Напротив, структурные модели требуют лишь определения строения программы безотносительно к результатам ее испытаний. По этой причине при оценке программной надежности ОС МАРС было принято решение руководствоваться моделями первого из указанных типов.

Структурные модели можно в свою очередь подразделить на два класса:

I) статические модели, рассматривающие в качестве элементов структуры программы простейшие операторы и операнды;

2) динамические модели, использующие представление программы в виде ориентированного графа. Вершинами графа являются линейные участки структуры программы, ребрами изображаются ветви передачи управления.

Статические методы отличаются большей простотой реализации, позволяя в то же время получать вполне приемлемые оценки надежности программного обеспечения. Далее рассматривается одна из таких моделей, предложенная в [4].

В качестве меры программной сложности здесь предлагается использовать показатель

$$C = (N_s + N_0)\log_2(n_s + n_0), \qquad (2)$$

где n_s – количество простых операторов программы, n_0 – количество простых операндов, N_s – количество повторений простых операторов, N_0 – количество повторений простых операндов.

В [4] также выдвинуто и обосновано для ряда конкретных систем программного обеспечения предположение о пропорциональности количества скрытых ошибок E сложности структуры программы:

$$E = \alpha C , \qquad (3)$$

где $\alpha = 3{,}3 \cdot 10^{-4}$. Соотношения (2), (3) могут быть использованы

для ориентировочной оценки качества программного обеспечения, планирования временных затрат на проведение опытной эксплуатации и оценки целесообразности внесения структурных изменений в разработанное программное обеспечение. Модель была проверена нами для 10 программ различного типа (вычислительных, сервисных, модулей ОС, служебных подпрограмм и др.). Обнаружено, что погрешность прогнозирования числа скрытых ошибок с помощью данной модели не превышает 30%.

Эксперименты показали также, что для определенных классов программ, выполняющих однотипные функции и обладающих сходной структурой, справедливо приближенное соотношение

$$C \approx \gamma D \ , \qquad (4)$$

где D - объем программы в машинных словах.

В качестве примера использования модели рассмотрим следующую задачу, возникшую при работе над программным обеспечением обмена двухмашинного вычислительного комплекса. В первоначальном варианте драйвер каждого внешнего устройства состоял из двух частей - блока анализа запросов и процедур управления. Поскольку блоки анализа всех драйверов обладали большой степенью общности, при разработке программного обеспечения обмена ОС МАРС было решено выделить этот блок, реализовав его на уровне ОС.

Пусть вычислительный комплекс включает N типов внешних устройств, обслуживаемых N драйверами. Тогда совокупная сложность подсистемы программного обеспечения управления обменом при децентрализованной обработке запросов составит

$$C_g = \sum_{i=1}^{N} (C_{0i} + C_i) \ ,$$

где C_{0i} - сложность блока анализа запросов i-го драйвера, C_i - общая сложность его же процедур управления. При наличии системного блока анализа запросов сложностью C_0 имеем совокупную сложность

$$C_c = C_0 + \sum_{i=1}^{N} C_i .$$

Введем показатель $\beta = C_g/C_c$, характеризующий степень уменьшения числа скрытых ошибок в программах обмена при внедрении схемы с системным блоком анализа запросов. Оценим верхнюю границу β при следующих условиях:

I) сложность блока анализа запросов любого из драйверов не превосходит сложности аналогичного системного блока: $C_{0i} \le C_0$, $i = 1,\dots, N$;

2) размеры системной области, отведенной для блока анализа запросов, ограничены объемом памяти D_0. Таким образом, из (4) следует $C_0 \le \gamma D_0$.

Введем параметры

$$C = \frac{\sum_{i=1}^{N} C_i}{N}, \qquad k = \frac{C_0}{C}.$$

Получим неравенство

$$\beta \le \frac{NC_0 + \sum_{i=1}^{N} C_i}{C_0 + \sum_{i=1}^{N}} = N\frac{C_0 + C}{C_0 + NC} = N\frac{k+1}{k+N}.$$

Для реального комплекса $N = 10$, $C = 4\cdot10^3$, $D_0 = 100$ словам. Кроме того, для драйверов внешних устройств ОС МАРС $\gamma \approx 19{,}0$. Отсюда получаем $k_{max} \approx 0{,}5$, т.е. $\beta_{max} \le 1{,}5$ и более чем полуторакратного увеличения надежности программного обеспечения при указанных ограничениях добиться невозможно.

Выводы

1. Практика показала, что на базе вычислительных средств АСВТ возможно построение высоконадежных вычислительных комплексов, обеспечивающих стабильное функционирование АСУ ТП сложных промышленных установок.

2. Для обеспечения надежной работы линии межмашинной связи в многомашинных вычислительных комплексах следует оптимизировать длину передаваемых по ним сообщений. При этом обмен большими массивами информации целесообразно вести через общую память на магнитных дисках.

В целях повышения надежности программы обеспечения многомашинного вычислительного комплекса необходима реализация стандартных модулей программы обмена на уровне ОС.

Литература

I. ЕВРЕИНОВ Э.В. Однородные вычислительные системы,структуры и среды. -М.: Радио и связь, 981. -208 с.

2. МАРТИН Дж. Системный анализ передачи данных.-Пер. с англ. -М.: Мир, 1975. Т.I. - 256 с.

3. ЛИПАЕВ В.В. Надежность программного обеспечения АСУ.-М.: Энергоиздат, 1981. - 240 с.

4. ХОЛСТЕД М.Х. Начала науки о программах. - Пер. с англ.-М.: Финансы и статистика, 1981. - 128 с.

Поступила в ред.-изд.отд.
21 июля 1982 года