Академія наук України
Інститут кібернетики імені В. М. Глушкова

На правах рукопису

ПАНФІЛЕНКО Валерій Павлович

УДК 681.3.06

БАГАТОРІВНЕВЕ ПРОЕКТУВАННЯ І ПЕРЕВІРКА ВЛАСТИВОСТЕЙ СТРУКТУРОВАНИХ ПАРАЛЕЛЬНИХ СХЕМ ПРОГРАМ

05.13.11 — математичне та програмне забезпечення обчислювальних машин, комплексів, систем та мереж

Автореферат дисертації на здобуття ученого ступеня кандидата фізико-математичних наук

Київ 1993

Роботу виконано в Інституті кібернетики імені В. М. Глушкова АН України.

Науковий керівник: доктор технічних наук ЦЕЙТЛІН Г. Є.

Офіційні опоненти: доктор фізико-математичних наук,
професор АНІСІМОВ А. В.,
кандидат фізико-математичних наук
ГОРОХОВСЬКІЙ С. С.

Провідна установа: Інститут програмних систем АН України.

Захист відбудеться «——» —————— 199 р. о ——— годині на засіданні спеціалізованої ради Д 016.45.01 при Інституті кібернетики імені В. М. Глушкова АН України за адресою:

252207 Київ 207, проспект Академіка Глушкова, 40.

З дисертацією можна ознайомитися в науково-технічному архіві інституту.

Автореферат розісланий «——» —————— 199 р.

Учений секретар
спеціалізованої ради СИНЯВСЬКИЙ В. Ф.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. В даний час інтенсивно розвиваються методи формалізованих специфікацій програм. В теорії програмування розроблено ряд формальних моделей і конструкцій, зручних для побудови строго обгрунтованих програмових систем поряд з доведенням наявності у них тих чи інших властивостей. До числа відомих методів відносяться системи логічного, алгебраїчного, композиційного, трансформаційного програмування, формалізовані технічні завдання, VDM, R-технологія і т.д. Розвиваються засоби інструментальної підтримки формалізованих специфікацій, в тому числі алгебраїчних. Відмітимо, що до числа перших вітчизняних мов алгебраїчного програмування відноситься АНАЛІТИК (Мир-2, СМ1410 та ін.), розроблений в Інституті кібернетики ім. В.М.Глушкова АН України. Набула відомості сучасна система алгебраїчного програмування APS, також розроблена в Інституті кібернетики.

Один з напрямів сучасного програмування, заснований на застосуванні алгебраїчних методів, сформувався як результат розвитку теорії систем алгоритмічних алгебр (САА) В.М.Глушкова. Для описання паралельних обчислень призначені модифіковані системи алгоритмічних алгебр (САА-М) . Апарат САА-М, який допускає компактне структурне подання схем програм у вигляді алгебраїчних формул, покладено в основу методу багаторівневого структурного проектування програм (БСПП), який розроблюється в Інституті кібернетики ім. В.М.Глушкова АН України. Метод БСПП орієнтований на побудову ієрархічних специфікацій класів алгоритмів і програм на основі поєднання алгебри алгоритмів і теорії формальних граматик. В рамках алгеброграматичного підходу формалізується проектування алгоритмів і програм як вивід в відповідних граматичних моделях. Семантика класів алгоритмів і програм визначається контекстно залежними правилами виводу і означенням нетермінальних символів в остаточних продукціях, що відповідає побудові інтерпретаційної алгебри. При цьому виникає проблема узгодженості рівнів проектування. Інструментальною підтримкою БСПП є структурний синтезатор програм МУЛЬТИПРОЦЕССИСТ, який за багаторівневими специфікаціями генерує програмовий продукт.

Зниження вартості розробки прямо пов'язане з відсутністю похибок при проектуванні. Побудова узгоджених специфікацій на різних рівнях абстракції і перевірка властивостей специфікацій на кожному рівні часто дозволяють виявити похибки на початкових етапах і, дякуючи цьому, уникнути повного перепроектування.

Мета і задачі роботи. Метою дисертації є:

– формалізація концепції узгодженості специфікацій алгоритмів і програм, заданих на різних рівнях абстракції, при проектуванні за методом БСПП;

– дослідження властивостей тупиковості (клінчів, деддлоків, зациклювань) і фіктивності (наявності компонент ітерації, які не виконуються) специфікацій паралельних програм;

– розробка для методу БСПП і його інструментарію засобів виявлення тупиків і фіктивностей у специфікаціях паралельних програм.

Наукова новизна. Для методу БСПП введено концепцію узгоджених специфікацій. Встановлені критерії узгодженості описів алгоритмів і програм на різних рівнях абстракції.

Досліджені властивості монотонності операторів у відповідних класах САА-М. Досліджена і алгоритмізована проблема тупиків і фіктивностей в асинхронних паралельних схемах програм.

Практична цінність. Концепція узгодженості специфікацій на різних рівнях абстракції має практичний інтерес при проектуванні і синтезі програм за методом БСПП у зв'язку з підвищенням надійності програмового продукту і прискоренням його налагодження.

Розширені засоби описання логічних умов для синхронізації паралельних процесів, що збагачує зображувальні можливості вхідної мови системи МУЛЬТИПРОЦЕССИСТ.

У рамках методу БСПП у системі МУЛЬТИПРОЦЕССИСТ реалізований аналізатор властивостей тупиковості і фіктивності в специфікаціях паралельних алгоритмів і програм. Сам аналізатор спроектований і синтезований засобами системи МУЛЬТИПРОЦЕССИСТ.

Дана робота виконувалась в рамках досліджень по темах:

– 1.13.4.5, N ГР01860045697 "Разработать теорию и методы автоматизированного производства проблемно-ориентированных систем";

- СИНТЕЗ "Дослідження математичних засад доказового програмування і трансформаційний синтез класів алгоритмів, програм і програмових систем".

Апробація. Результати роботи доповідались на 7-му (Київ, 1986) і 8-му (Київ, 1988) семінарах "Паралельне програмування і високопродуктивні системи", на семінарах наукової ради АН України з проблеми "Кібернетика".

Публікації. По темі дисертації опубліковано 6 робіт.

Структура роботи. Дисертаційна робота складається з вступу, чотирьох розділів, заключення і додатків.

ЗМІСТ РОБОТИ

У вступі обгрунтовується актуальність роботи, сформульована її мета, показана наукова новизна, дано короткий зміст дисертації по розділах.

Перший розділ присвячено узгодженню формалізованих специфікацій при багаторівневому проектуванні класів алгоритмів і програм на базі систем алгоритмічних алгебр.

САА - двохосновна алгебра, яка складається з алгебри умов і алгебри операторів з узагальненими булевими (диз'юнкція, кон'юнкція, заперечення, прогнозування) і традиційними структурними (композиція, α-диз'юнкція (альтернатива), α-ітерація (цикл)) операціями. Оператори - це відображення деякої множини (яка зветься інформаційною) в себе, а умови - відображення інформаційної множини в множину логічних констант.

Метод БСПП основується на апараті кінцевопороджених САА, оператори яких являють собою алгоритми і програми на семантичному рівні. Для представлення на синтаксичному рівні використовуються терми - слова в алфавіті, які є записом правильно побудованих виразів над символами предметних змінних САА. Формальна семантика задається відображенням алгебри термів в однотипну з нею САА. Значення терма можна отримати як результат виконання над значеннями операторних і логічних символів всіх тих операцій, які відповідають функціональним символам у записі терма.

Для задання класів алгоритмів і програм на синтаксичному рівні в методі БСПП призначені граматики структурного проектування (ГСП) - контекстно залежні граматики, в яких термінальний алфавіт складають функціональні символи з сигнатури і предметні змінні САА. Слова, які виводяться в ГСП, – це операторні терми. Пара ГСП і САА може служити формалізованою специфікацією класу алгоритмів і програм. Вибір конкретної САА задає ступінь деталізації (рівень абстракції), з якою представляються алгоритми і програми.

В розділі 1 розглядаються специфікації програм на різних рівнях абстракції. Визначаються поняття порівневого переходу і зв'язок специфікацій програм на різних рівнях. Перехід з рівня на рівень полягає в уточненні і конкретизації програми. Багаторівневе проектування розуміється як розробка програми в напрямку зниження рівня абстракції.

Запропонована концепція узгодженості специфікацій алгоритмів і програм на різних рівнях абстракції, яка базується на розділенні синтаксичного і семантичного аспектів проектування, прийнятому в алгебраїчному програмуванні. Узгодженість специфікацій тісно пов'язана з запропонованим у дисертації поняттям конкретизації. Це дозволяє сформулювати критерії узгодженості специфікацій алгоритмів і програм на основі їх коректної конкретизації.

Зниження рівня абстракції безпосередньо пов'язане зі зміною структури інформаційної множини САА, оскільки більш "подрібнена" структура інформаційної множини здатна задавати більш "детальні" відображення. Позначимо через A і A' САА з більш високим і низьким рівнями абстракції відповідно. Тим самим побудова A' може бути досягнута "розподілом" елементів інформаційної множини A на цілі класи нових більш подрібнених елементів.

Відповідність між елементами інформаційних множин породжує відношення Φ між елементами САА A' і A. Елементи A', які відповідають елементам A, звуться уточненнями. Кожний елемент A має, можливо, кілька уточнень в A', але не кожний елемент A' може бути уточненням деякого елемента з A. Вибираючи для кожного елемента з A по одному уточненню з A', дістаємо конкретизацію елементів A в A'.

Теорема 1. Множина уточнень елементів з A утворює підалгебру A'; обернене відображення Φ^{-1} є гомоморфізмом з A' в A, а конкретизація є ізоморфним вкладенням A в A'.

Позначимо через T і T' алгебри термів, відповідні A і A', а через h і h' - інтерпретуючі відображення з T в A і з T' в A' відповідно. Назвемо уточненням терма $t \in T$ терм $t' \in T'$, який дістаємо підстановкою деяких припустимих термів з T' замість предметних змінних з t. Конкретизацією термів $\hat{\rho}$ будемо називати взаємно однозначну відповідність термів з T і підмножини їх уточнень з T'.

Теорема 2. Конкретизація $\hat{\rho}$ є ізоморфним вкладенням T в T'.

Конкретизації $\hat{\rho}_1$ і $\hat{\rho}_2$ термів назвемо семантично еквівалентними, якщо $\forall t' \in T'$ $h(\hat{\rho}_1(t'))=h(\hat{\rho}_2(t'))$. Відношення семантичної еквівалентності дійсно буде еквівалентністю на множині конкретизацій термів. Конкретизації термів $\hat{\rho}_1$ і $\hat{\rho}_2$ будуть семантично еквівалентні тоді і лише тоді, коли $h(\hat{\rho}_1(t'))=h(\hat{\rho}_2(t'))$, $i=\overline{1,n}$ для предметних змінних з T'.

Конкретизації термів і елементів САА відповідають синтаксичному і семантичному аспектам проектування. Специфікацією класу алгоритмів на заданому рівні абстракції назвемо трійку <G,A,h>, де G - ГСП, A - САА, h - інтерпретуюче відображення з алгебри термів T в A. Нехай A має більш високий рівень абстракції, ніж A', тоді <G,A,h> і <G',A',h'> специфікації алгоритмів на різних рівнях абстракції. Специфікації <G,A,h> і <G',A',h'> назвемо узгодженими, якщо з того, що $t'=\hat{\rho}(t)$, виходить $h(t)=\phi(h'(t'))$, $t \in T, t' \in T'$. Це означає, що узгоджені специфікації, як можливі конкретизації термів з T, припускають не всі уточнення з T', а лише ті, інтерпретації яких в САА A' є уточненням відповідних елементів САА A.

Пару $\langle\rho,\hat{\rho}\rangle$ назвемо конкретизацією специфікацій, якщо ρ - конкретизація элементів з A, а $\hat{\rho}$ - конкретизація термів з $L(G) \subseteq T$. Конкретизацію $\langle\rho,\hat{\rho}\rangle$, назвемо коректною, якщо $\forall t \in T$ $h'(\hat{\rho}(t))=\rho(h(t))$. Нехай $\hat{\rho}$ конкретизація термів. Тоді, якщо найдеться така конкретизація ρ елементів САА, що конкретизація $\langle\rho,\hat{\rho}\rangle$ специфікацій коректна, то вона єдина. І навпаки, для відомої конкретизації ρ елементів САА існує така конкретизація $\hat{\rho}$ термів, що конкретизація $\langle\rho,\hat{\rho}\rangle$ специфікацій коректна.

Теорема 3. Специфікації на різних рівнях абстракції узгоджені тоді і лише тоді, коли конкретизація специфікацій коректна.

Таким чином, узгоджені специфікації і коректні конкретизації зводяться одні до одних.

Конкретизації $\langle\rho,\hat{\rho}_1\rangle$ і $\langle\rho,\hat{\rho}_2\rangle$ специфікацій назвемо семантично еквівалентними, якщо $\hat{\rho}_1$ і $\hat{\rho}_2$ семантично еквівалентні. Відмітимо, що всякі коректні конкретизації вигляду $\langle\rho,\hat{\rho}_1\rangle$ і $\langle\rho,\hat{\rho}_2\rangle$ семантично еквівалентні. Тоді множина коректних конкретизацій розпадається на класи по відношенню семантичної еквівалентності. Для будь-якого ρ існує єдина (з точністю до семантичної еквівалентності) $\hat{\rho}$ така, що конкретизація $\langle\rho,\hat{\rho}\rangle$ коректна. Таким чином, коректна конкретизація $\langle\rho,\hat{\rho}\rangle$ визначається однозначно з ρ з точністю до семантичної еквівалентності.

Терми $t_1,t_2\in T$ назвемо семантично еквівалентними (при інтерпретації h), якщо $h(t_1) = h(t_2)$. Тоді коректна конкретизація перетворює семантично еквівалентні терми в семантично еквівалентні.

Нехай ГСП G' задає клас алгоритмів $L(G) \subseteq T$ і ГСП G' задає клас алгоритмів $L(G') \subseteq T'$ на більш низькому рівні абстракції. Тоді для $L(G) \subseteq T$ $\hat{\rho}(L(G)) \subseteq T'$ будуть складати конкретизації алгоритмів на більш низькому рівні абстракції.

Зміну заданої ГСП G на ГСП G' при проектуванні алгоритма назвемо порівневим переходом, якщо

1). $\forall t\in L(G)\ \rho(h(t)) = h'(\hat{\rho}(t))$.

2). $\hat{\rho}(L(G)) = L(G')$.

Умова 1 виходить з коректності конкретизації. Упорядкована по включенню множина мов $L\subseteq T'$, що задовольняють умові 1, складають повну решітку, де $\hat{\rho}(L(G))$ буде єдиним найменшим елементом. Тоді умова 2 гарантує єдиність побудови $L(G')$.

Нехай $G=\langle T,N,\sigma,P\rangle$ і $G'=\langle T',N',\sigma',P'\rangle$ -ГСП і $\hat{c}_i$, $\hat{c}_i'$, $i=\overline{1,m}$ - предметні змінні T і T' відповідно. За побудовою $\rho(\hat{c}_i)=t_i'(\hat{c}_1',\ldots,\hat{c}_m')$, $i=\overline{1,m}$.

<u>Теорема 4</u>. При порівневому переході ГСП G' можна отримати з ГСП G за наступними правилами:

$$T' = \{\hat{c}_i',\ i=\overline{1,m}\},\ N' = N \cup T,\ \sigma' = \sigma,$$

$$P' = P \cup \{\hat{c}_i \longrightarrow t_i'(\hat{c}_1',\ldots,\hat{c}_m'),\ i=\overline{1,n}\}.$$

Послідовність порівневих переходів назвемо багаторівневим проектуванням. Воно означає послідовне розширення ГСП при зниженні рівня абстракції пов'язаної з ним САА. Таким чином, мова йде про послідовне уточнення класу алгоритмів, які проектуються.

Теорема 5. Послідовність коректних конкретизацій є коректною конкретизацією, і семантично еквівалентні терми при коректній конкретизації перетворюються в семантично еквівалентні.

Розділи 2,3 присвячені перевірці властивостей тупиковості і фіктивності паралельних регулярних схем програм (ПРС).

Розглядаються модифіковані САА, орієнтовані на формалізацію паралельних обчислень. Множина логічних значень в алгебрі умов доповнена значенням "невизначеність" для опису аварійних ситуацій. Сигнатура САА-М включає операцію розділеної диз'юнкції, яка полягає в асинхронному паралельному виконанні операторів. Тоді паралельні регулярні схеми - це операторні терми. Синхронізація обчислень в паралельних регулярних схемах досягається за допомогою розставлення контрольних точок і операторів затримки (очікування), з ними пов'язаних. Контрольною точкою називається місце на стику операторів ПРС. З контрольною точкою пов'язана логічна умова, яка стає істиною, коли процес обчислень досягає даної контрольної точки. Припускається, що логічна умова продовжує залишатися істинною і надалі у процесі обчислень. Така властивість зветься властивістю замкнености. Логічна умова, яка має цю властивість, зветься замкненою. Синхронізатором (оператором затримки, очікування) називається оператор, який виконує затримку обчислень до моменту досягненості процесом обчислень контрольної точки, йому відповідної. Шляхом тотожних перетворень ПРС можна представити в стандартизованому вигляді, коли, зокрема, α-ітерації не будуть утримувати внутрішніх контрольних точок.

В другому розділі розглядаються системи алгоритмічних алгебр з замкненими логічними умовами. Досліджується замкненість логічних виразів. Доведені твердження про збереження властивості замкненості при логічних операціях і про розкладення умов і виразів на областях обчислень. Разповсюдження властивості замкненості на складні логічні умови і області обчислень, які не перетинаються, розширює засоби синхронізації паралельних процесів на вказаних областях.

Нехай пара <S,F> - система алгоритмічних алгебр і M - інформаційна множина (множина станів). Припускається, що в M існує невизначений стан $w \in M$, такий що $\forall a \in S_{оп}$ $a(w)=w$. Підмножина $M' \subseteq M$ зветься замкненою (відносно операцій САА), якщо $\forall a \in S_{оп}$ $\forall q \in M'$ $a(q) \in M'$. Підмножина $M' \subseteq M$ зветься ізольованою (відносно операцій САА), якщо її доповнення $M \setminus M'$ замкнене.

Твердження 1. Якщо $M_1 \subseteq M$, $M_2 \subseteq M$ і M_1, M_2 - замкнені (ізольовані), то $M_1 \cup M_2$ і $M_1 \cap M_2$ - замкнені (ізольовані).

Твердження 2. Замкнені (ізольовані) підмножини інформаційної множини M складають решітку.

Нехай $M^0(\alpha) = \{ q \in M \mid \alpha(q)=0 \}$ - множина хибності,

$M^{\mu}(\alpha) = \{ q \in M \mid \alpha(q)=\mu \}$ - множина невизначеності,

$M^1(\alpha) = \{ q \in M \mid \alpha(q)=1 \}$ - множина істинності.

Логічна умова $\alpha \in S_{усл}$ замкнена, якщо $M^1(\alpha)$ і $M^{\mu}(\alpha)$ - замкнені.

Твердження 3. Якщо $\alpha, \beta \in S_{усл}$ і α, β - замкнені, то кон'юнкція $\alpha \wedge \beta$ - замкнена.

Диз'юнкція замкнених умов, взагалі кажучи, не замкнена.

Твердження 4. Достатні ознаки замкненості диз'юнкції.

1. Якщо α, β замкнені і всюди визначені ($M^{\mu}(\alpha)=\varnothing$ і $M^{\mu}(\beta)=\varnothing$), то $\alpha \vee \beta$ - замкнене.

2. Якщо α, β замкнені і $M^1(\alpha) \cap M^{\mu}(\beta)=\varnothing$ і $M^{\mu}(\alpha) \cap M^1(\beta)=\varnothing$, то $\alpha \vee \beta$ - замкнене.

Нехай $M = \bigcup_{i=1}^{n} M_i$, $M_i \cap M_j=\varnothing$, $i \neq j$, де M^i замкнене $\forall i=\overline{1,n}$.

Замкненість M_i означає, що $\forall a \in S_{оп}$ і $\forall q \in M_i$ $a(q) \in M_i$ $\forall i \in \overline{1,n}$.

Введемо такі позначення:

$M_i^1(\alpha) = \{ q \in M_i \mid \alpha(q) = 1 \}$,

$M_i^0(\alpha) = \{ q \in M_i \mid \alpha(q) = 0 \}$,

$M_i^{\mu}(\alpha) = \{ q \in M_i \mid \alpha(q) = \mu \}$.

Будемо говорити, що логічна умова α замкнена на $M_i \in M$, якщо $M_i^1(\alpha)$ і $M_i^{\mu}(\alpha)$ замкнені.

Твердження 5. Якщо α замкнена на M, то α замкнена на M_i, $i=1,n$, і навпаки, якщо α_i замкнені на M_i, $i=\overline{1,n}$, то умова

$$\alpha(q) = \begin{cases} \alpha_1(q), \text{ якщо } q \in M_1 \\ \dots \\ \alpha_n(q), \text{ якщо } q \in M_n \end{cases}, \text{ замкнена на множині } M.$$

Довизначимо умову α_1 так, щоб вона була замкнена як на M_1, так і на М. Нехай α логічна умова, визначена на М. Логічні умови

$$\alpha_i^{\eta}(q) = \begin{cases} \alpha(q), \text{ якщо } q \in M_i \\ \eta \quad , \text{ якщо } q \notin M \end{cases}, \ \eta \in \{0, \mu, 1\}$$

назвемо звуженням умови α на M_i, $i=\overline{1,n}$. Тоді умова α може бути представлена таким чином: $\alpha = \bigvee_{i=1}^{n} \alpha_i^0 = \bigwedge_{i=1}^{n} \alpha_i^1$ і замкнена тоді і лише тоді, коли $\forall i=\overline{1,n}$ α^1 (α^0) - замкнені.

Вірні такі рівності:

$$\alpha \wedge \beta = \bigwedge_{i=1}^{n} (\alpha_i^1 \wedge \beta_i^1) = \bigvee_{i=1}^{n} (\alpha_i^0 \wedge \beta_i^0), \ \alpha \vee \beta = \bigwedge_{i=1}^{n} (\alpha_i^1 \vee \beta_i^1) = \bigvee_{i=1}^{n} (\alpha_i^0 \vee \beta_i^0).$$

Слідує відмітити справедливість наступних співвідношень:

$\{a\}_{\overline{\alpha}} = \alpha$, $(a \vee b)_{\overline{\alpha}} = \llcorner\alpha\lrcorner t \uplus \llcorner\overline{\alpha}\lrcorner a$, де α - замкнена логічна умова, $\llcorner\alpha\lrcorner = \{E,N\}_{\alpha}$ - операція фільтрації, E - тотожній, N - невизначений оператори. В силу приведенных співвідношень, а також на підставі тверджень 1-5 і розробленой в теорії САА-М процедури стандартизації ПРС справедлива наступна теорема: довільна ПРС у САА-М з замкненими логічними умовами зводиться до форми, в якої логічні умови в α-диз'юнкціях і α-ітераціях подані монотонними ДНФ.

Збереження властивості замкненості при логічних операціях і розкладення умов і виразів на областях обчислень дозволяють використовувати логічні вирази від замкнених логічних умов для синхронізації паралельних процесів, що збагачує зображувальні можливості вхідної мови системи МУЛЬТИПРОЦЕССИСТ.

Третій розділ присвячений перевірці властивостей тупиковості і фіктивності ітеративних структур у специфікаціях паралельних програм. Розглядаються представлення схем паралельних програм у вигляді паралельних регулярних схем в САА з замкненими логічними умовами. Досліджується проблема тупиків (зациклювань) і фіктивностей (компонент ітерації, які не виконуються) при синхронізації виконання послідовних процесів за допомогою контрольних точок і очікувань. Проведена класифікація тупиків, установлено критерії тупиковості і фіктивності в ПРС, доведена дуальність задач виявлення тупиків і фіктивностей. Пропонується алгоритм виявлення тупиків і фіктивностей, який має лінійну залежність від числа контрольних точок і очікувань.

Розглянемо абстрактну модель обчислювальної системи, що є узагальненням абстрактної моделі ЕОМ і являє собою композицію операційної і управляючої структур. Операційна структура поділяється на підструктури $P=\{P^i | i\in I\}$, які не перетинаються. Управляюча структура являє собою сукупність управляючих терміналів $U=\{U^i | i\in I\}$, що діють у своїх активних зонах: P^i - активна зона U^i $\forall i\in I$.

Щоб мати можливість відобразити факт одночасного незалежного виконання операторів a і b в їх активних зонах, до сигнатури САА додається операція розділеної диз'юнкції. Розділена діз'юнкція операторів a і b являє собою одночасне паралельне виконання операторів a і b. Нехай A - САА з замкненими логічними умовами і T_A - відповідна їй алгебра термів. Тоді паралельні регулярні схеми (ПРС) - це операторні терми з T_A.

Синхронізація обчислень в паралельних регулярних схемах здійснюється шляхом розстановки контрольних точок і операторів затримки (очікування), з ними пов'язаних.

Нехай $t=0,1,\ldots$ означає дискретний час. Контрольною точкою зветься місце на стику операторів, що входять у схему, яка пов'язується з замкненою логічною умовою.

$$\alpha(t)=\begin{cases} 1, & \text{якщо процес обчислень до моменту } t \text{ досяг контрольноъ точки,} \\ 0, & \text{якщо процес обчислень до моменту } t \text{ не досяг контрольноъ точки,} \\ \mu, & \text{якщо процес обчислень аварійно завершений до моменту досягнення контрольноъ точки.} \end{cases}$$

Умова α зветься умовою синхронізації.

Синхронізатором (оператором затримки, очікування) зветься оператор $S_i(\alpha)=\{E_i\}_\alpha$, де E_i - тотожній оператор з активною зоною P_i. Синхронізатор здійснює затримку обчислень в активній зоні до моменту досягнення процесом обчислень контрольної точки, з якою пов'язана дана умова синхронізації.

Шляхом застосування тотожніх співвідношень паралельну регулярну схему можна представити в стандартизованому вигляді, коли α-ітерації не будуть вміщувати внутрішніх контрольних точок.

В результаті ПРС може бути приведена до форми, яка складається з паралельних гілок, а паралельна гілка - це розділена диз'юнкція послідовних гілок.

Оператори ПРС, контрольні точки і ітерації утворюють помічені вершини графа, який назвемо ПРС-графом. Вершини, що належать до однієї послідовної гілки, зв'язуються дугами - попередня вершина з наступною. Додатково з кожної контрольної точки виходять дуги, які ведуть до відповідних ітерацій.

Якщо у ПРС-графі існує цикл (замкнений шлях), то всі ітерації, що входять у нього, - тупикові. Вони називаються тупиковими операціями першого роду, а зв'язані з ними контрольні точки - тупиковими точками першого роду.

Теорема 6. Паралельна гілка тупикова тоді і лише тоді, коли існує хоча б одна тупикова ітерація першого роду.

Таким чином, доведено, що кожна тупикова паралельна гілка включає цикл і, як наслідок, тупикові ітерації першого роду істотні, тобто не можуть бути усунені з тупикової паралельної гілки (інакше гілка перестає бути тупиковою).

Послідовність пар контрольних точок і відповідних ітерацій, зв'язаних дугами, утворює ланцюг, якщо вона починається з тупикової ітерації першого роду. Всі ітерації, які входять у ланцюг, тупикові, і вони звуться ітераціями другого роду. Якщо вилучити тупикові ітерації першого роду і при цьому не виникнуть нові тупикові ітерації першого роду, то тупикові ітерації другого роду не будуть тупиковими. Отже показано, що тупикові ітерації другого роду не істотні , тобто не будуть тупиковими, якщо не буде тупиків першого роду.

Теорема 7. Існують тупикові ітерації лише першого і другого родів і їх множини не перетинаються.

Схеми синхронізації, які дістаємо одну з другої шляхом перестановки контрольної точки і ітерації, назвемо дуальними. Відмітимо, що ітерація тупикова тоді і лише тоді, коли вона фіктивна у дуальній схемі синхронізації. Отже, задачу пошуку фіктивностей можна звести до пошуку тупиків у дуальній схемі синхронізації і навпаки.

Процес обчислень у ПРС моделюється функціонуванням відповідної мережі Петрі. Мережу Петрі дістаємо з ПРС-графа моделюванням вузлів графа місцями і переходами мережі. Початкова розмітка полягає у розміщенні фішок у перші оператори послідовних гілок. Кінцева розмітка має фішки в останніх операторах послідовних гілок. Побудовою забезпечується однакова синхронізація обчислень у ПРС і відповідній їй мережі Петрі.

Теорема 8. ПРС буде тупиковою тоді і лише тоді, коли відповідна їй мережа Петрі припиняється не у кінцевій розмітці. Ітерації, які мають фішки - тупикові.

Щоб мати можливість характеризувати стан ПРС і відповідної їй мережі Петрі, що з'являються під час обчислень, вводяться відношення попередності (<) і поняття досягненості у процесі обчислень.

Твердження 6. Ітерація I(α) - тупикова першого роду тоді і лише тоді, коли I(α)<T(α).

Твердження 7. Ітерація I(α) - фіктивна тоді і лише тоді, коли T(α)<I(α).

Результати теорем 6-8 і тверджень 6, 7 використовуються при проектуванні алгоритму виявлення тупиків і фіктивностей у ПРС.

Четвертий розділ присвячений розвитку інструментарію структурного синтезу програм, спроектованих за методом БСПП. Основу інструментарію складає система МУЛЬТИПРОЦЕССИСТ - структурний синтезатор програм. Вхідним для синтезатора служить САА/1 - мова проектування в структурованих схемах програм (САА-схемах), які базуються на апараті САА-М. Система МУЛЬТИПРОЦЕССИСТ доповнена аналізатором тупиків і фіктивностей в специфікаціях паралельних алгоритмів і програм. Аналізатор моделює проходження у процесі обчислень контрольних точок і ітерацій у ПРС. При цьому тупики будуть виявлені, коли обчислення по всіх гілках будуть затримані (теорема 8, твердження 6).

Неформальний опис алгоритма виявлення тупиків.

1. Вилучення контрольних точок, які досягнені.
2. Вилучення синхронізаторів, які досягнені.
3. Якщо схема синхронізації пуста, то паралельна гілка безтупикова, інакше досягнені ітерації - тупики.

Фіктивну ітерацію процес обчислень завжди пройде до контрольної точки, яка з нею зв'язана (твердження 7), що також буде відмічено при роботі алгоритма. Врахування особливостей ПРС-графа дозволяє побудувати алгоритм виявлення тупиків і фіктивностей лінійної складності O(n), де n - число вузлів графа (число контрольних точок і очікувань).

Аналізатор розроблений і реалізований засобами системи МУЛЬТИПРОЦЕССИСТ. Наведені САА-схеми аналізатора.

Основні результати дисертації

Виконана дисертаційна робота присвячена формалізації узгодженості специфікацій алгоритмів і програм при проектуванні за методом БСПП, дослідженню властивостей тупиковості і фіктивності паралельних регулярних схем програм і реалізації інструментальних засобів для їх перевірки. Робота включає такі основні результати:

1. Запропоновано концепцію узгодженості, що базується на розділенні синтаксичного і семантичного аспектів проектування. Встановлено критерії узгодженості специфікацій алгоритмів і програм на різних рівнях абстракції. На основі отриманих критеріїв запропоновано розширення граматики структурного проектування при переході до специфікацій на більш низькому рівні абстракції.

2. Доведено твердження про збереження властивості замкненості при логічних операціях і розкладенні умов і виразів на областях обчислень. Розширено засоби задання паралелізму, які дозволяють використання виразів над замкненими логічними умовами для синхронізації паралельних процесів.

3. Досліджена проблема тупиків (зациклювань) і фіктивностей (компонент ітерації, які не виконуються) при синхронізації виконання послідовних гілок за допомогою контрольних точок і очікувань. Проведено класифікацію тупиків, установлено критерії тупиковості і фіктивності в ПРС, доведено дуальність задач виявлення тупиків і фіктивностей. На основі отриманих результатів розроблений алгоритм виявлення тупиків і фіктивностей.

4. Система МУЛЬТИПРОЦЕССИСТ доповнена аналізатором тупиків і фіктивностей в специфікаціях паралельних алгоритмів і програм. Аналізатор має лінійну складність від числа контрольних точок і очікувань.

Основні положення дисертації опубліковано в таких роботах:

1. Панфиленко В. П. О формальных основах многоуровневого проектирования // Тез. докл. 7-й Всесоюз. шк.-семинара «Параллельное программирование и высокопроизводительные системы». — Киев : Ин-т кибернетики им. В. М. Глушкова АН УССР, 1986. — С. 80—82.

2. Нагорная Л. И., Панфиленко В. П. Поуровневое проектирование транслятора с параллельного КОБОЛа // Средства реализации систем программирования. — Киев : Ин-т кибернетики им. В. М. Глушкова АН УССР, 1987. — С. 8—14.

3. Панфиленко В. П. О грамматиках структур [illegible] в системах алгоритмических алгебр в многоуровневом проектировании // Тез. докл. 8-го Всесоюз. семинара «Параллельное программирование и высокопроизводительные системы». — Киев : Ин-т кибернетики им. В. М. Глушкова АН УССР, 1988. — С. 32—34.

4. Панфиленко В. П. Алгоритм проверки невырожденности алгебры термов // Средства представления знаний в информационных технологиях. — Киев : Ин-т кибернетики им. В. М. Глушкова АН УССР, 1992. — С. 60—64.

5. Панфиленко В. П. Согласованные спецификации при многоуровневом проектировании программ // Разработка математического и программного обеспечения ППП и решение задач дискретной оптимизации. — Киев : Ин-т кибернетики им. В. М. Глушкова АН УССР, 1992. — С. 89—95.

6. Панфиленко В. П. Тупики в параллельных регулярных схемах программ. — Киев, 1993. — 23 с. — (Препр. / АН Украины. Ин-т кибернетики им. В. М. Глушкова ; 93-1).

Підп. до друку 23.03.93. Формат 60×84/16. Папір друк. №2. Офс. друк. Ум. друк. арк. 0,70. Ум. фарбо-відб. 0,82. Обл.-вид. арк. 1,0. Тираж 100 прим. Зам. 539.

Редакційно-видавничий відділ з поліграфічною дільницею Інституту кібернетики імені В. М. Глушкова АН України
252207 Київ 207, проспект Академіка Глушкова, 40