

УДК 004.896; 621.7.01; 658.511.3; 658.58; 658.52; 658.589

О.Д.Криськов, проф., канд. техн. наук, І.М. Іваник, асп., К.К.Щербина, магістр
Кіровоградський національний технічний університет

Особливості взаємозв'язку полів при проектуванні переходів в САПР регламентів технологічного процесу на основі ООП

На основі методики об'єктно-орієнтованого програмування (ООП) висвітлено зв'язки між полями, що використовуються в процесі проектування елементарного регламенту технологічного процесу, що складається з однієї операції в складі одного технологічного переходу. Розглянуто поля різного рівня доступності, визначено спільні поля та поля-властивості: для батьківського класу `TexprocesClass`, дочірнього класу `TokenOperacClass` та похідного від класу технологічна операція – `PidrizTorClass`. Запропоновано алгоритм отримання техніко-економічної оцінки на різних етапах проектування.

регламент технологічного процесу, техніко-економічна оцінка, поля, поля-властивості, ООП

Вступ. На даному етапі розвитку промисловості важливою справою є своєчасність розробки економічних, вигідних для даного підприємства в даний час регламентів технологічних процесів (РТП) виготовлення деталей. Кожна хвилина затримки приводить до економічних втрат. Втрати різко зростають, якщо для впровадження у виробництво вибрати економічно не обґрунтований варіант РТП. Для прискорення процесу проектування шляхом автоматизації рутинних процесів використовують програмні продукти типу “Автопроект”, “SWR-Технологія” [1,2] тощо. Проте при їх розробці методиці економічної оцінки прийманих технологом рішень уваги приділялось недостатньо. Між тим, питання про економічну доцільність технологічних рішень є одним із наріжних каменів теорії технології машинобудування. Рішенню цього питання перешкоджає не тільки складність вирішення задачі, а й рухомість в часі та в просторі (тобто від підприємства до підприємства) як основних техніко-економічних показників РТП так і значень коефіцієнтів та величин, покладених в основу такої оцінки. Не останньою причиною є відсутність базових автоматизованих методик. Питання значно спрощується якщо поєднати методику об'єктного проектування РТП з біжучою оцінкою технологічних рішень на основі нормативного методу техніко-економічного аналізу (ТЕА) технологічних рішень.

Метою дослідження є висвітлення структури алгоритму проектування, що дозволяє уникнути дублювання інформації та паралельних чи повторних алгоритмів при виконанні технологічних та економічних обчислень, дякуючи певній структуризації процесу проектування, на основі ідей наслідування, інкапсуляції та інших особливостей програмування в середовищі, що підтримує ООП.

Основна частина. Досягнення поставленої мети демонструється через певний механізм отримання поточної техніко-економічної оцінки прийнятого рішення на рівні переходу в процесі динамічного аналітичного проектування РТП на прикладі найпростішого варіанта технологічного процесу, що складається з однієї операції (токарної) в складі якої є один перехід (підрізка торця). При цьому пропонуються певні схеми організації взаємодії та взаємозв'язку простих полів та полів-властивостей об'єкту ПЕРЕХІД як в самому об'єкті, так і з батьківськими полями об'єктів ОПЕРАЦІЯ та ТЕХНОЛОГІЧНИЙ ПРОЦЕС. При черговій модернізації САПРТ РТП `TECHNOL` [3], спираючись на ООП, на базі `TokenOperacClass` для кожного з технологічних переходів може бути створено свій клас, наприклад, клас `PidrizTorClass`. Згідно з принципами ООП батьківський щодо переходу клас `TokenOperacClass` є похідним від класу `TexprocesClass`. Класу перехід притаманні певні поля, що необхідні для відповідного опису алгоритму моделювання фізичних та економічних

© О.Д.Криськов, І.М. Іваник, К.К.Щербина, 2010

процесів. Для прикладу, звернемо увагу на поля, які відображають зміст робіт, притаманних для ряду поширених одноінструментальних токарних (свердлильних) операцій. Далі на основі класу `PidrizTorClass` створюємо об'єкт ПЕРЕХІД, перелік полів якого представлені в таб.1. Серед інших є поле символічного коду типового переходу, яке відображає його назву, поле для розміщення операційного ескізу (рис. 1), що графічно відображає зміст технологічного переходу тощо.

Розглянемо взаємозв'язки полів дочірнього класу `PidrizTorClass`, що описують вищезгадану технологічну операцію з батьківськими класами `ToкарOperacClass` та `TexprocesClass` (рис.1). Необхідні поля цих класів зведені до таблиць 1,2, що підкреслює структуру їх взаємозв'язку і дозволяє чіткіше висвітлити доступність та взаємозв'язок полів.

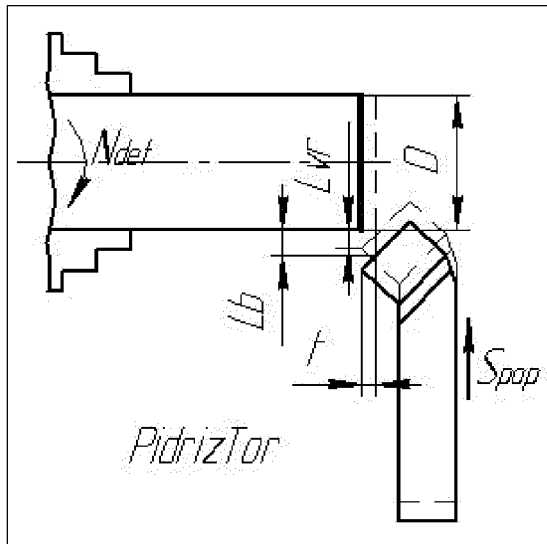


Рисунок 1 – Підрізати торець прутка діаметром D

Розглянемо спочатку особливості доступу до простих полів, притаманних вищезгаданим класам, позначеним в табл. 1,2 однією зірочкою (*). Ряд таких полів класу `TexprocesClass` технолог-проектант заповнює в інтерактивному режимі, наприклад, вказуючи шифр та назву деталі, її матеріал, тип заготовки та її механічні властивості, тип виробництва, вартість 1 кВт/год електроенергії тощо. Другі прості поля користувач заповнює в `ToкарOperacClass` в діалоговому режимі при визначенні змісту чергової операції. До таких полів віднесемо назву операції, модель верстата, назву чи код кріпильного чи контрольного пристосувань і таке інше.

Проектуючи зміст наступної технологічної операції у діалоговому режимі та орієнтуючись на стилізовані графічні зображення, відповідні змісту типових технологічних переходів, користувач конструює зміст технологічної операції, як послідовність типових переходів. При цьому він додатково заповнює ряд простих полів, зокрема, параметри, що конкретизують типові переходи до рівня індивідуальних, притаманних саме цій деталі-операції чи деталі-переходу. До таких полів віднесемо, геометричні розміри оброблюваної поверхні, марку ріжучої частини інструменту, тип обробки (чорнова, напівчистова) тощо.

В подальшому, опираючись на методику ООП, та програмуючи певні розрахунки, притаманні тому чи іншому переходу, використовуючи можливості наслідування, інкапсуляції та обумовлені цими технологіями методики доступу до полів об'єктів різних класів ми можемо в об'єкті ПЕРЕХІД забезпечити доступ по простих полів знизу вверх по ієрархії підпорядкування об'єктів ОПЕРАЦІЯ та МАРШРУТ, створених на базі батьківських класів (`ToкарOperacClass` та `TexprocesClass`). Програмно це забезпечується описом полів, як змінних в процедурах типу `SetAll`, наприклад:

```
Procedure SetAll (Name Det, ShufrDet, CodeMaterialDet, PrizvuscheRozrob, PrizvuscheNach, PrizvuscheZavKafedru, TermoObrob, Galvanic, NumberChex, XarakterZagot, TupVurob, DateRozrob:string; GroupVerst, ModelVerstat, NameOper, Prisposob, Vudprivod, KontPrispo, SposobNorm:String; FVerst, P, KiltDvug, KiltYRE, VartistOblad, rTstuch1:real; sTransition1:string; S, V, NewN, Tosn, Lraz, D, t, Lbez, Lvri, Ldov, Tshtuch:real; Certfcate_MachineOborot1, CodeInstr, GOST, XarakInstr, sTupObr1:string; rEnergia, rVutrRijInstr, rZatrNaInstrument, rSigma, rHb, rKrock, rB, rKutVplani, rTstiy:real; bPeretoch:byte); overload.
```

Таблиця 1 – Поля PerehodovClass

Зміст поля		Ідентифікатор поля	Доступ	Тип поля	Класи переходів						
					PidrizTorClass	ProtPovClass	SverdOtvClass	VidrizDetClass	FrezPazKinClass		
Ріжучий інструмент	Найменування	sNameInstr	pb	*							
	ГОСТ (ДСТУ)	sGOSTInstr	pb	*							
	Особливі характеристики ін-та		sXarakInstr	pb	*	+	+	+	+	+	
	Код	Символьний	sCodeInstr	pb	*	+	+	+	+	+	
		Числовий	bCodeInstr	pb							
	Період стійкості, хв		rTstiy	pb	*	+	+	+	+	+	
	Кількість переточок		bPeretoch	pb	*	+	+	+	+	+	
	Головний кут в плані, ф		rKutVplani	pb	*	+	+	+	+	+	
Діаметр інструмента (поверхні), мм		rD	pb	*							
Обробляема поверхня	Глибина різання, мм		rt	pv	*	+	+	+	+	+	
	Величина врізання, мм		rLvr	pb	*	+	+	+	+	+	
	Величина перебігу, мм		rLp	pb							
	Довж.безударного підводу, мм		rLbp	pb	*	+	+	+	+	+	
	Довж.обробляем. поверхні, мм		rLk	pb	*	+	+	+	+	+	
Параметри реж. різання	нормативні	Подача	мм/об	rS_normativ	pv	**	+	+	+	+	+
			мм/зуб	rSz	pv	**	-	-	-	-	+
	Швидкість різання	м/хв		rV_normativ	pv	**					
		Оберти		об/хв	rNroz	pv	**				
	фактичні	Подача	мм/об	rS_fact	pv	**	+	+	+	+	+
			мм/зуб	rSz_fact	pv	**	-	-	-	-	+
			мм/хв	rSx_fact	pv	**					
		Швид. різання	м/хв	rVfact	pv	**					
Оберти	об/хв	rN_ObPasp	pv	**							
Основний час, хв		rTosn	pv	**	+	+	+	+			
Контрольний інструмент	Найменування	sNameKistr	pb	*							
	ГОСТ (ДСТУ)	sCodeKistr	pb	*							
Витрати, віднесені до переходу, на	Експлуатацію приміщення		rVutr_p_P	pv	**	+	+	+	+	+	
	Оплату послуг робітника		Vposl_r_P	pv	**	+	+	+	+	+	
	Оплату електроенергії		Venerg_P	pv	**	+	+	+	+	+	
	Обладнання	Амортизація	Vamor_P	pv	**	+	+	+	+	+	
		Ремонт	Vrem_P	pv	**	+	+	+	+	+	
		Утримання	Vutr_o_P	pv	**	+	+	+	+	+	
	Пристосування		Vpris_P	pv	**	+	+	+	+	+	
	Контрольний інструмент		Vpris_k_P	pv	**	+	+	+	+	+	
Ріжучий інструмент		Vins_rij_P	pv	**	+	+	+	+	+		

Примітка: поля з доступом private – pv, public – pb; * - просте поле; ** – поле-властивість (поле-член).

Візьмемо, для прикладу, просте поле sNameInstr, у якому буде записано назва ріжучого інструмента. Основними факторами при виборі останнього є форма оброблюваної поверхні, форма та траєкторія руху формоутворюючого леза інструмента відносно заготовки, які в сукупності визначають зміст переходу. Останній в системі TEXNOL закодовано в виді

певного символічного коду. Очевидно, що в багатьох випадках одну і ту ж поверхню можна обробити з прийнятними показниками якості різними інструментами. При цьому може навіть змінитися структура операції чи навіть структура РТП. В кожному разі виникає очевидне питання, яка з можливих структур буде економічно вигіднішою.

Особливістю полів-властивостей, позначених в табл. 1 та 2 двома зірочками, є те, що до них прямого доступу користувач не має. Вони інкапсульовані (простіше скажемо сховані) в дочірніх класах, а доступ до них єдиний можливий через спеціальні методи Set та Get. Алгоритм останніх передбачає певні аналітичні розрахунки в залежності від змісту процедур, які побудовані на основі алгоритмів відомих з курсів теорії різання, проектування ріжучих інструментів, технології машинобудування, металорізальних верстатів тощо.

Нижче приведено фрагмент коду, який забезпечує доступ до об'єкта поля класу PidrizTorClass з типом захисту private з використанням методів Get та Set.

```
function PidrizTorClass.GetsTransition:ts;
begin
  GetsTransition := Technol.sTransition;
end;
procedure PidrizTorClass.SetsTransition(sTransition1:string);
begin
  Technol.sTransition := sTransition1;
end;
```

Стосовно Get та Set. Зазвичай це префікси в назві процедур, які дозволяють прив'язати певні методи до яких буде використовуватися той чи інший алгоритм. Решта назви процедури це вже прив'язка до алгоритму проекту.

Другим прикладом поля-властивості, об'явленим з типом доступу private, є поле відповідальне за основний час, величина якого може змінюватися лише в конкретному об'єкті ПЕРЕХІД з використанням методу **SetTosn** згідно коду:

```
function ProtPovClass.GetTosn:real;
begin
  GetTosn := Technol.rTosn;
end;
procedure ProtPovClass.SetTosn(Tosn:real);
begin
  Technol.rTosn := Tosn;
end;
```

Таблиця 2 – Поля класів OperacionClass та TexprocesClass

Зміст поля		Ідентифікатор поля	Доступ	Тип поля	
Поля OperacionClass	Назва операції	sNameOper	pv	**	
	Назва верстата	Група	sGroupVerst	pv	**
		Модель	sModelMechines	pv	**
	Виробнича площа під верстат	rFverst	pv	**	
	Потужність двигунів верстата, кВт	Головного	rNd	pv	**
		1 допоміжного	rNd1	pv	**
		2 допоміжного	rNd2	pv	**
	Кількість одиниць ремонтної складності	Механічної	rRSM	pv	**
		Електричної	rRSE	pv	**
		Електронної	rRSEl	pv	**
	Балансова вартість обладнання, грн	rVartistOblad	pb	**	
	Затискне пристосування	Код	bZatuskPrispos	pb	*
Назва		sZatuskPrispos	pb	*	

Контрольне пристосування	Код	bKontrPrispos	pb	*	
	Назва	sKontrPrispos	pb	*	
Витрати в грн., віднесені до операції, на	Експлуатацію приміщення	rVutr_p_O	pv	*	
	Оплату послуг робітників	rVposl_r_O	pv	*	
	Електроенергію	rVenerg_O	pv	*	
	Амортизацію обладнання	rVamor_O	pv	*	
	Ремонт обладнання	rVrem_O	pv	*	
	Утримання обладнання	rVutr_o_O	pv	*	
	Пристосування	rVpris_O	pv	*	
	Контрольний інструмент	rVpris_k_O	pv	*	
	Ріжучий інструмент	rVins_rij_O	pv	*	
	Технологічна вартість операції	rTechnoSobivartOper	pv	*	
Поля ТехпроцесClass	Тип виробництва		sTupVurob	pb	**
	Матеріал деталі		sCodeMaterialDet	pb	**
	Група (оброблюваності) матеріалу		sCodeGroupMaterialDet	pv	**
	Розробник		sPrizvuscheRozrob	pb	*
	Нормоконтроль		sNormoKontrol	pb	*
	Затверджую		sPrizvuscheNach	pb	*
	Назва	Деталі	sNameDet	pb	*
	Шифр		sShufrDet	pb	*
	Характеристика та механічні властивості заготовки	Тип	boXarakterZagot	pb	**
		Особливості	bStanPov	pb	**
		Маса	rMasa	pb	**
		Міцність	rSigma	pb	**
		Твердість	rHb	pb	**
	Вартість 1 кВт*год. електроенергії, грн		rEnergia	pv	**
	Витрати в грн. віднесені до операції на	Експлуатацію приміщення	rVutr_p_M	pv	**
		Оплату послуг робітників	rVposl_r_M	pv	**
		Оплату електроенергію	rVenerg_M	pv	**
		Амортизацію обладнання	rVamor_M	pv	**
		Ремонт обладнання	rVrem_M	pv	**
		Утримання обладнання	rVutr_o_M	pv	**
Пристосування		rVpris_M	pv	**	
Контрольний інструмент		rVpris_k_M	pv	**	
Ріжучий інструмент		rVins_rij_M	pv	**	
Технологічна вартість деталі		rTechSobivart	pv	**	

Примітка поля з доступом private – pv, public – pb; * - просте поле; ** - поле-властивість (поле-член).

Основний час є величиною розрахунковою, яка визначається за певним алгоритмом автоматично в об'єкті ПЕРЕХІД на основі даних, приведених як в простих так і в полях-властивостях. Отже між цими полями існують певні алгоритмічні зв'язки, які, для прикладу, у загальному вигляді можна записати так:

$$Tosn_{P_i} = \frac{Lroz}{Sxv}, \text{ хв.}, \quad N_i = \frac{P_z v}{102 \cdot 60}; \text{ кВт}, \quad Q_i = \frac{Tosn_{P_i} \cdot N}{\cdot 60}, \text{ кВт*час.}$$

Отже в той час, коли між полями $Tosn_{P_i}$, N_i та Q_i об'єкту ПЕРЕХІД існує вище приведений алгоритмічний зв'язок, то на рівні ОБ'ЄКТУ операція існує очевидний зв'язок і необхідність сумування основного часу та витрат електроенергії на перехід. Ці та очевидні зв'язки економічного характеру (трудомісткість та технологічна вартість переходу і операції в цілому) інкапсульовані в методах типу SetTosn_Pi та GetTosn_Pi, забезпечують цілісність

та актуальність даних в процесі моделювання структури технологічної операції, внаслідок автоматичного перерахування по інкапсульованим алгоритмам при зміні значень простих полів. При допомозі методу `property` опис здійснюється так:

```
property rTshtuch:real read GetrTshtuch write SetrTshtuch;
```

Поле властивості сили різання `rPz` з полем доступом `public`, який залежить від заданих параметрів глибини різання, подачі та іншого ряду показників. Значення поля змінюється один раз в кожному об'єкті з використанням лиш методу `SetrPz`, за кодом:

```
procedure PidrizTorClass.SetrPz(rPz:real);  
begin  
  rPz:=rPz;  
end;
```

Висновок. Запропонована ідеологія проектування переходів та операцій РТП на основі ООП та приведено алгоритм побудови її структури. Досліджена організація взаємодії простих полів та полів-властивостей, різних класів: батьківського, дочірнього та субдочірнього в процесі ДАПР та біжучої ТЕО, що дозволяє визначити економічну доцільність використання технологічного переходу в одноінструментальній операції.

Список літератури

1. Криськов О.Д., Щербина К.К. ООП та методика проектування регламенту технологічного процесу. Загальнодержавний міжвідомчий науково-технічний збірник. Конструювання, виробництво та експлуатація сільськогосподарських машин, вип.. 39 – Кіровоград: КНТУ, 2009. - 448с.
2. Картавов С.А. Технология машиностроения / специальная часть / : Учебник для машиностроительных спец. вузов, 2-е изд., перераб. и доп. – К.: Вища школа, 1984. – 272с.
3. Криськов О.Д. Система автоматизированого оформлення регламентов технологических процессов на ЭВМ ЕС для серийного многономенклатурного производства Республиканский межведомственный научно-технический сборник /Автоматизация производственных процессов / Вып. 23.,1984.-Львов: Вища школа, С.119-126

О. Криськов, И. Иваньк, К. Щербина

Особенности взаимосвязи полей при проектирование переходов в САПР регламентов технологического процесса на основе ООП

На основе методики объектно-ориентированного программирования (ООП) отражены связи между полями, которые используются в процессе проектирования элементарного регламента технологического процесса, который состоит из одной операции в составе одного технологического перехода. Рассмотрены поля разного уровня доступности, определенно общие поля и поля-свойства: для родительского класса `Texprocesclass`, дочернего класса `Токарoperacclass` и производного, от класса технологическая операция – `Pidriztorclass`. Предложен алгоритм получения технико-экономической оценки на разных этапах проектирования.

О. Kriskov, I. Ivanyk, K. Scherbina

Features of intercommunication of the fields at planning of passing to CAD of regulations of technological process on the basis of OOP

On the basis of method of the object-oriented programming (OOP) copulas are reflected between the fields, which are used in the process of planning of elementary regulation of technological process which consists of one operation in composition of one technological transition. The fields of different level of availability are considered, certainly the common fields and fields-properties: for the paternal class of `Texprocesclass`, daughter's class of `Токарoperacclass` and derivative, from a class technological operation – `Pidriztorclass`. The algorithm of receipt of *tekhniko-ekonomichnoy* estimation is offered on the different stages of planning.

Одержано 21.01.10