

ALEKSANDER MATUSZAK\*

ALGORYTM WYZNACZANIA CZĘŚCI ŚCISKANEJ  
PRZEKROJU PRĘTA ORAZ OBLICZANIA  
JEJ CHARAKTERYSTYK GEOMETRYCZNYCHALGORITHM FOR DETERMINING COMPRESSED REGION  
OF CROSS-SECTION AND COMPUTING  
ITS MOMENTS OF AREA

## Streszczenie

W wielu przypadkach koncepcje, które są wykorzystywane do budowy algorytmu komputerowego różnią się istotnie od koncepcji wykorzystywanych w metodach analitycznych. W artykule zebrano i podsumowano te koncepcje, które służą do budowy efektywnego algorytmu obliczania charakterystyk geometrycznych części ściskanej przekroju prętowego. Przeanalizowano zamianę całki powierzchniowej na liniową, twierdzenie Greena oraz opis brzegu obszaru. Sformułowano parę algorytmów: jeden służący do obliczania charakterystyk geometrycznych oraz drugi do wyznaczania strefy ściskanej. Pokazano prosty przykład pozwalający prześledzić opisywane algorytmy.

*Słowa kluczowe: przekrój żelbetowy, charakterystyki geometryczne, całkowanie naprężeń*

## Abstract

Algorithm for computer can be very different from sequence of analytical calculs. In this paper the necessary background for efficient algorithm for computing compressed region moments of area is analysed. Conversion from area integral to one dimensional integral, Green's theorem and boundary representation of cross-section help to formulate the pair of algorithms: one for computing moments of area and the second for determining compressed region of cross-section. Simple numerical example that allow one to analyse steps of algorithm is provided.

*Keywords: RC cross-section, moments of area, stress integration*

\* Dr inż. Aleksander Matuszak, Instytut Technologii Informatycznych w Inżynierii Lądowej, Wydział Inżynierii Lądowej, Politechnika Krakowska.

## 1. Wstęp

Przedmiotem artykułu są dwa, częściowo powiązane, algorytmy mające zastosowanie przy rozwiązywaniu przekroju: algorytm wyznaczania strefy ściskanej oraz algorytm wyznaczania charakterystyk geometrycznych obszaru.

Zadanie wyznaczania strefy ściskanej przekroju pręta występuje przede wszystkim w konstrukcjach żelbetowych, gdzie modelowo pomija się pracę betonu w strefie rozciąganej. Analogiczne zadanie pojawia się jednak w zaskakująco wielu innych zadaniach projektowych. W artykule skupiono się na modelu naprężeń liniowych (NL) dla żelbetu, jako najprostszym, w którym występują wszystkie elementy ogólnego zadania: podział przekroju na podobszary oraz obliczanie wszystkich typowych charakterystyk przekroju. Jednak należy to traktować jedynie jako przykład ogólnego zadania.

Zadanie wyznaczania strefy ściskanej może być rozumiane jako: a) dla zadanych sił przekrojowych, b) dla zadanej deformacji przekroju. To drugie może prowadzić do pierwszego, jeśli będzie fragmentem iteracyjnego procesu poszukiwania deformacji zapewniającej równowagę. Tutaj rozważany jest tylko drugi wariant tego zadania.

W obu przypadkach zadanie to nie jest zagadnieniem samodzielnym ale jest potrzebne do obliczenia wielkości całkowitych związanych z przekrojem; typowo sił wewnętrznych i pokrewnych wartości. Z uwagi na skalę zastosowań, temu stosunkowo prostemu, ale często żmudnemu zagadnieniu poświęca się wiele uwagi.

W konsekwencji konieczność rozwiązania tego zadania pojawia się często w oprogramowaniu pisany samodzielnie, na własne potrzeby, przez inżynierów. Wydaje się, że brak jest powszechnej świadomości jak takie zadanie należy rozwiązać. Stosowanie metod używanych w obliczeniach analitycznych nie prowadzi do satysfakcjonującego algorytmu. Prosty i skuteczny algorytm rozwiązania jest bardzo odmienny od metod używanych w obliczeniach analitycznych. Stąd zachodzi potrzeba, aby w miarę dokładnie i przystępnie zebrać i opisać wszystkie elementy niezbędne do uzyskania dobrego algorytmu.

Przy budowie algorytmu wyznaczania wielkości przekrojowych możliwe są dwa podejścia. Jedno proste koncepcyjnie, ale wymagające użycia algorytmów geometrycznych służących do jawnego wyznaczenia strefy ściskanej, oraz drugie, znacznie bardziej złożone, ale niemal eliminujące operacje na geometrii (Zupan [16], rozwinięcie Bonet [2]). Opisany tutaj algorytm jest raczej adresowany do inżynierów niż do programistów. Dlatego uwaga zostanie skupiona na tym pierwszym, jako prostym koncepcyjnie i łatwym w implementacji, nawet za cenę nieco mniejszej ogólności i wydajności.

Dla przedziałowo-wielomianowych modeli materiału wielkości przekrojowe wygodnie jest obliczać przez charakterystyki geometryczne odpowiedniej części przekroju. Dla NL istotne są jedynie charakterystyki części ściskanej. W przypadku bardziej skomplikowanych modeli materiału może istnieć potrzeba podziału przekroju na podobszary wedle bardziej złożonych kryteriów. Mogą wtedy też wystąpić charakterystyki wyższych rzędów. Przykładowo, Sousa [14] i Charalampakis [4] sprowadzają całkowanie sił przekrojowych dla betonu opisanego parabolą madrycką do charakterystyk geometrycznych rzędu trzeciego.

W podejściu wykorzystującym charakterystyki zadanie rozpada się na dwie części: wyznaczenie części ściskanej i znalezienie charakterystyk geometrycznych tejże części. W artykule odwrócono kolejność, najpierw rozważa się zagadnienie obliczania charakterystyk

całego przekroju, a następnie zagadnienie wyznaczenia strefy ściskanej. Użycie odpowiedniego opisu geometrii do obliczania charakterystyk skutkuje bardzo prostym algorytmem wyznaczenia części ściskanej.

Charakterystyki geometryczne przekroju są postrzegane jako zamknięte zagadnienie. Są albo stabilizowane (np. profile walcowane), albo są sumą prostych figur; najczęściej prostokątów. W przypadku strefy ściskanej – a więc jedynie części przekroju – znajdowanie charakterystyk się komplikuje, gdyż tablice są mało użyteczne, a części ściskane prostokątów stają się np. trapezami. Dla strefy ściskanej prostokąta daje się znaleźć względnie proste wyrażenia (Silva [5], Cedolin [3], Izzudin [8], Bonet [1]), ale nie wszystkie przekroje są prostokątne.

Użycie całek liniowych (brzegowych) zamiast powierzchniowych jest sposobem wyjścia poza przekroje prostokątne bez konieczności dyskretyzacji obszaru, czy to jawnie w postaci prostokątów (di Ludovico [6], Zak [15]), czy niejawnie poprzez zliczanie pixeli (Sfakianakis [12]). W przypadku oprogramowania nie bez znaczenia jest fakt, że całki liniowe istotnie redukują koszt (czas) obliczeń.

Całki liniowe w obliczeniach wielkości przekrojowych są wykorzystywane dosyć powszechnie, ale czasami jest to nieczytelne bądź przemieszane z innymi zagadnieniami. Już w skrypcie Palucha [11] z 1988 roku jest zamieszczony program do obliczania charakterystyk, który stosuje całki liniowe zamiast powierzchniowych, jednak używane wzory są jedynie podane w tekście programu, bez wyprowadzenia czy wyjaśnienia. Kawakami [9] podaje wzory całkowania jawnie ale nie bardzo przejrzyście, więc Bonet [2] pisze, że „są trudne do implementacji” a di Ludovico [6] bezkrytycznie powtarza tę opinię.

Do zamiany całek powierzchniowych na liniowe można wykorzystać całkę iterowaną bądź twierdzenie Greena. To ostatnie akcentuje istotny element – reprezentację brzegu.

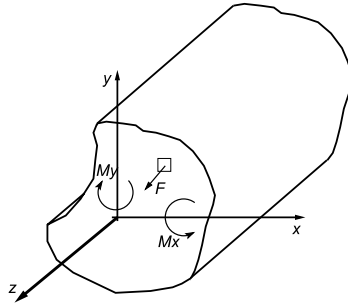
Kwestie związane z brzegiem nie wydają się być wszystkim znane i zupełnie lekceważone. Algorytmy geometryczne operujące na brzegu najczęściej są opisywane dosyć pobieżnie [14], czasami opis sprowadza się wręcz do kilku ogólnikowych zdań [4]. Na podstawie takich opisów trudno zrealizować program. Opis, reprezentacja i operacje na brzegu obszaru okazują się być kluczowe dla zbudowania uniwersalnego algorytmu. Przy odpowiednim opisie brzegu algorytm wyznaczenia strefy ściskanej staje się dość prostą konsekwencją tego opisu brzegu. Co więcej, reprezentacja części ściskanej jest analogiczna do reprezentacji całego przekroju. Jeśli jest dostępna procedura wyznaczenia charakterystyk przekroju opisanego w takiej reprezentacji, to automatycznie dostaje się możliwość obliczenia charakterystyk części ściskanej przekroju.

Stąd kwestie brzegu i reprezentacji danych opisujących brzeg pokazane zostaną bardzo szczegółowo i zajmują dużą część miejsca w artykule.

## 2. Zginanie przekroju pręta smukłego

Rozważany jest przekrój zelbetowego pręta smukłego opisywanego według teorii naprężeń liniowych.

Wszystkie równania zapisywane będą w układzie współrzędnych pokazanym na rys. 1. Oś pręta jest osią  $z$ . Początek układu współrzędnych nie musi być w środku ciężkości przekroju. Osie  $x$ ,  $y$  nie muszą być główne.



Rys. 1. Układ współrzędnych i znakowanie wielkości przekrojowych  
Fig. 1. Coordinates system and cross-section forces

Uwzględnia się jedynie naprężenie normalne  $\sigma = \sigma_{zz}$  i także odkształcenie  $\varepsilon = \varepsilon_{zz}$ . Za dodatnie uważa się naprężenia (siły) rozciągające.

Zginanie jest rozpatrywane w ramach teorii pręta smukłego Bernoulliego-Eulera. Przy takim założeniu płaszczyna odkształceń jest opisana równaniem:

$$\varepsilon(x, y) = \varepsilon_0 + y\kappa_x + x\kappa_y \quad (1)$$

gdzie:

$\varepsilon_0$  – odkształcenie w początku układu współrzędnych

Krzywizny w (1) definiowane są jako pochodne kąta obrotu:

$$\kappa_x = \frac{\partial \varphi_x}{\partial y}, \quad \kappa_y = \frac{\partial \varphi_y}{\partial x} \quad (2)$$

Znakowanie momentów zostało tak przyjęte, że dodatnie naprężenie w pierwszej ćwiartce układu wywołuje dodatnie momenty. Kąty obrotu są tak samo znakowane jak momenty (por. rys. 1).

Obciążenia ograniczają się do dowolnej kombinacji momentów zginających i siły podłużnej. Zakłada się, że siły poprzeczne i moment skręcający nie wpływają na naprężenia normalne, więc są pomijane.

Model konstytutywny dla betonu wg NL to materiał liniowy nie przenoszący rozciągania:

$$\sigma(\varepsilon) = \begin{cases} E^b \varepsilon & \varepsilon < 0 \\ 0 & \varepsilon \geq 0 \end{cases}$$

Dla stali zakłada się znacznie prostsze prawo fizyczne:  $\sigma = E^s \varepsilon$ .

Znając naprężenia w każdym punkcie przekroju  $\sigma(\varepsilon(x, y))$ , można obliczyć zestaw sił przekrojowych (w początku układu współrzędnych).

$$N = \int_A \sigma(x, y) dA, \quad M_x = \int_A \sigma(x, y) y dA, \quad M_y = \int_A \sigma(x, y) x dA \quad (3)$$

Po podstawieniu równań fizycznych do równań (3), rozdzieleniu obszaru całkowania najpierw na powierzchnię betonu  $A_b$  oraz pozostałą  $A - A_b$ , a następnie po uwzględnieniu, że naprężenia w betonie nie znikają jedynie w części ściskanej  $A_p$ , dostaje się układ równań ([10])

$$\begin{bmatrix} E^b A_d + E^s A_s & E^b S_x^b + E^s S_x^s & E^b S_y^b + E^s S_y^s \\ E^b S_x^b + E^s S_x^s & E^b J_x^b + E^s J_x^s & E^b D_{xy}^b + E^s D_{xy}^s \\ E^b S_y^b + E^s S_y^s & E^b D_{xy}^b + E^s D_{xy}^s & E^b J_y^b + E^s J_y^s \end{bmatrix} \begin{bmatrix} \varepsilon_0 \\ \kappa_x \\ \kappa_y \end{bmatrix} = \begin{bmatrix} N \\ M_x \\ M_y \end{bmatrix} \quad (4)$$

gdzie charakterystyki geometryczne dla przekroju betonowego z indeksem górnym  $b$  dotyczą części ściskanej betonu  $A_d$ . Oprócz charakterystyk powierzchni występują charakterystyki zbrojenia, z indeksem górnym  $s$ . Może to być zarówno zbrojenie miękkie sprowadzane do punktów, jak i zbrojenie sztywne (kształtowniki). Charakterystyki stali nie będą tutaj omawiane. Charakterystyki dla betonu w (4) obliczane są dla części ściskanej, co sprawia, że nie są cechą przekroju ale również zależą od stanu deformacji. W przypadku poszukiwania stanu równowagi nie można ich policzyć raz na początku. należy je obliczać dla każdego położenia osi obrotu.

Natomiast dla znanej (zadanej) części ściskanej  $A_d$  obliczanie charakterystyk nie różni się od obliczania charakterystyk dla całego przekroju. Na obliczaniu charakterystyk dla zadanego obszaru zostanie obecnie skupiona uwaga.

### 3. Całki brzegowe

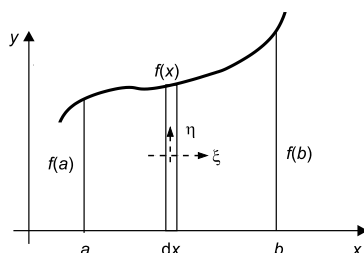
Wszystkie charakterystyki geometryczne przekroju można zdefiniować jako całki z wielomianu  $c_{n,m}(x,y) = x^n y^m$  w obszarze przekroju.

$$C_{n,m} = \int_A C_{n,m}(x,y) dA = \int_A x^n y^m dA \quad (5)$$

Zamiast obliczać takie całki jako całki powierzchniowe można wprost wyrazić je jako całki liniowe, korzystając z zasad mechaniki, co może być interesujące dla inżynierów. Wykorzystane zostaną jedynie wzory na charakterystyki geometryczne dla prostokąta oraz twierdzenie Steinera.

#### 3.1. Obszar pod krzywą

Obszar przekroju, pokazany na rys. 2, jest ograniczony od góry krzywą o równaniu  $f(x)$ , osią  $x$  oraz prostymi  $x = a$ ,  $x = b$ .



Rys. 2. Obszar z brzegiem zadanym funkcją  $f(x)$

Fig. 2. Domain bounded by function  $f(x)$

Rozumowanie jest następujące: rozważamy elementarny wycinek (prostokąt) pod krzywą (rys. 2) w punkcie o rzędnej  $x$ . Wartość funkcji (wysokość prostokąta) to  $f(x)$ . Długość podstawy to  $dx$ . Więc  $dA = f(x) \cdot dx$ . Sumując (całkując) te wartości w przedziale od  $a$  do  $b$ , otrzymuje się pole powierzchni:

$$A = \int_A 1dA = \int_a^b f(x)dx \quad (6)$$

Dla momentu statycznego względem osi  $x$  rozumowanie jest analogiczne. Moment statyczny elementarnego prostokąta  $dS_x$  względem  $x$  jest polem powierzchni pomnożonym przez odległość od środka ciężkości, która wynosi  $f(x)/2$ , więc  $dS_x = f(x)dx \cdot f(x)/2$ . Całkując, dostaje się:

$$S_x = \int_A dS_x = \int_a^b \frac{1}{2} f(x) \cdot f(x)dx = \frac{1}{2} \int_a^b f^2(x)dx \quad (7)$$

Analogicznie można obliczyć moment statyczny względem osi  $y$ , mnożąc pole powierzchni  $f(x) \cdot dx$  przez odległość od osi, która jest dana wprost jako rzędna  $x$ .

$$S_y = \int_a^b x \cdot f(x)dx = \int_a^b xf(x)dx \quad (8)$$

Dla prostokąta  $B \times H$  moment bezwładności, względem osi  $\xi$  przechodzącej przez środek ciężkości prostokąta wynosi  $J_\xi = \frac{BH^3}{12}$ . Z tw. Steinera wynika, że moment bezwładności względem osi  $x$  wynosi  $J_x = J_\xi + A\Delta y^2$ , gdzie  $\Delta y$  jest odległością środka ciężkości od osi  $x$ .

$$dJ_x = \frac{1}{12} f^3(x)dx + f(x)dx \cdot \left(\frac{1}{2} f(x)\right)^2 \quad (9)$$

$$J_x = \int_A dJ_x = \frac{1}{3} \int_a^b f^3(x)dx$$

Moment dewiacji  $D_{xy}$  można obliczyć korzystając z faktu, że jego wartość w środku ciężkości prostokąta, w układzie osi głównych centralnych  $\xi, \eta$  jest zero, a z tw. Steinera wynika  $D_{xy} = D_{\xi\eta} + A\Delta x\Delta y$ .

$$D_{xy} = \int_a^b \left(0 + f(x)dx \cdot x \frac{1}{2} f(x)\right) = \frac{1}{2} \int_a^b xf^2(x)dx \quad (10)$$

Dla momentu bezwładności  $J_y$ , jeśli wyrazi się przyrost wartości:

$$dJ_y = \frac{1}{12} (dx)^3 \cdot f(x) + f(x) \cdot dx \cdot x^2$$

a następnie pominię wyraz  $(dx)^3$  jako mały, pozostaje  $dJ_y = x^2 f(x)dx$ , całkując, otrzymuje się:

$$J_y = \int_a^b x^2 f(x)dx \quad (11)$$

### 3.2. Brzeg odcinkowo-liniowy

Poprzednio wyprowadzone wzory (6)–(11) są ważne dla dowolnej krzywej opisującej brzeg dającej się wyrazić jako funkcja zmiennej  $x$ .

Z punktu widzenia praktyki bardzo ważny jest przypadek, kiedy brzeg jest opisany łamaną. Wtedy na każdym fragmencie brzegu  $f(x)$  jest funkcją liniową. Wzory się upraszczają i stają się na tyle wygodne, że nawet brzeg krzywoliniowy (np. odcinek łuku kołowego) przybliżany jest łamaną. Charalampakis [4] pokazuje – na dość skrajnym przykładzie – że mało dokładne przybliżenie łamaną może generować istotny błąd rozwiązania. Obecność zakrzywionych – szczególnie krótkich – fragmentów brzegu utrudnia zadanie.

W przypadku brzegu opisanego łamaną podział na fragmenty będzie odpowiadał podziałowi przekroju na trapezy.

Dla trapezu funkcja  $f(x)$  jest funkcją liniową. Dla uproszczenia, wartości funkcji na końcach przedziału będą oznaczane jako  $f(a) = f_a$  na początku oraz  $f(b) = f_b$  na końcu (rys. 2). Wtedy równanie krzywej można zapisać:

$$f(x) = f_a + \frac{f_b - f_a}{b - a} \cdot (x - a) \quad (12)$$

Podstawiając równanie brzegu (12) do kolejnych wyrażeń (6)–(11), można otrzymać odpowiednie wzory, jednak prowadzi to do bardzo żmudnych rachunków.

Aby tego uniknąć, wystarczy najpierw zauważyć, że wszystkie funkcje podcałkowe w równaniach (6)–(11) są wielomianami stopnia co najwyżej trzeciego, a następnie skorzystać ze wzoru Simpsona.

Wzór Simpsona na całkę z funkcji  $g(x)$  wyraża się jako:

$$\int_a^b g(x) dx = \frac{1}{6} (g_a + 4g_p + g_b) (b - a) \quad (13)$$

gdzie:

$g_a, g_b$  – wartości funkcji na końcach przedziału,  
 $g_p$  – wartość funkcji w środku przedziału całkowania.

Jest to wzór dokładny dla wielomianu stopnia nie wyższego niż trzeci.

Wzór (13) po wstawieniu  $g(x) = f(x)$  wyraża pole powierzchni, ale dla funkcji liniowej (12) zachodzi:  $f_p = \frac{1}{2}(f_a + f_b)$  oraz  $x_p = \frac{1}{2}(x_a + x_b)$ , więc wzór upraszcza się do znanego wzoru na pole powierzchni trapezu.

Dla momentu statycznego, traktując, że dana jest funkcja  $g(x) = f^2(x)$ , wzór Simpsona daje:

$$S_x = \frac{1}{2} \int_a^b f^2(x) dx = \frac{1}{12} (f_a^2 + 4f_p^2 + f_b^2) (b - a) \quad (14)$$

Po uwzględnieniu liniowości  $f(x)$  wzór (14) można przepisać jako:

$$S_x = \frac{1}{6} (f_a^2 + f_a f_b + f_b^2) (b - a) \quad (15)$$

Podobnie przebiega całkowanie momentu statycznego względem osi  $y$ :

$$S_y = \int_a^b xf'(x)dx = \frac{1}{6}(a \cdot f_a + 4x_p \cdot f_p + b \cdot f_b)(b-a) \quad (16)$$

Po uwzględnieniu liniowych zależności dla  $x_p$  i  $f_p$  dostaje się:

$$S_y = \frac{1}{6}(2a \cdot f_a + a \cdot f_b + b \cdot f_a + 2b \cdot f_b)(b-a) \quad (17)$$

Dla momentu bezwładności funkcją podcałkową jest  $f^3(x)$  – więc traktując to jako całkowanie funkcji  $g(x) = f^3(x)$ , dostaje się:

$$J_x = \frac{1}{18}(f_a^3 + 4f_p^3 + f_b^3)(b-a) \quad (18)$$

Po uwzględnieniu liniowej zależności  $f_p$  od  $f_a$  i  $f_b$  dostaje się:

$$J_x = \frac{1}{12}(f_a^3 + f_a^2 f_b + f_a f_b^2 + f_b^3)(b-a) \quad (19)$$

Moment dewiacji można wyprowadzić w analogiczny sposób:

$$D_{xy} = \frac{1}{2} \int_a^b xf^2(x)dx = \frac{1}{12}(a \cdot f_a^2 + 4x_p \cdot f_p^2 + b \cdot f_b^2)(b-a) \quad (20)$$

Po uwzględnieniu liniowości członów dostaje się:

$$D_{xy} = \frac{1}{24}(a(3f_a^2 + f_b^2 + 2f_a f_b) + b(3f_b^2 + f_a^2 + 2f_a f_b))(b-a) \quad (21)$$

Analogicznie postępując z  $J_y$ :

$$J_y = \frac{1}{6}(a^2 f_a + 4x_p^2 f_p + b^2 f_b)(b-a) \quad (22)$$

a po uwzględnieniu liniowości członów:

$$J_y = \frac{1}{12}(f_a(3a^2 + 2ab + b^2) + f_b(a^2 + 2ab + 3b^2))(b-a) \quad (23)$$

W ten względnie łatwy sposób dostaje się zestaw wzorów na charakterystyki geometryczne dla trapezu.

Kawakami [9] wyprowadza równoważne wzory, ale wyraża je poprzez  $y_a$  i  $\Delta y$ , a następnie rozwija potęgę dwumianu.

Kwadratura Simpsona – używana zazwyczaj do przybliżonego całkowania – została tu użyta jedynie do uproszczenia przekształcania wzorów; wyniki są równoważne wyprowadzeniom analitycznym. Natomiast użycie tutaj kwadratury numerycznej tworzy naturalne przejście między całkowaniem analitycznym – co jest równoważne użyciu dokładnej kwadratury – a całkowaniem przybliżonym, czyli użyciem kwadratury niższego rzędu niż wymagana bądź użycia kwadratur dokładnych jedynie dla wielomianów (Newton–Cotes, Gauss) do całkowania funkcji nie będących wielomianami.



Równoważnie można użyć dwupunktowej kwadratury Gaussa. Jednak jej użycie w tym przypadku nie ma specjalnego uzasadnienia. Dokładność obu jest identyczna. Pozornie kwadratura Gaussa jest bardziej wydajna, gdyż wymaga dwu punktów całkowania a kwadratura Simpsona trzech. Jednak końcowy punkt całkowania jednego odcinka pokrywa się z początkowym następnego, więc jeśli w ogóle wydajność jest istotna, to można oprogramować zadanie, używając efektywnie dwu punktów na odcinek.

Przybliżone całkowanie jest osobnym zagadnieniem i wykracza poza ramy tego artykułu.

### 3.3. Brzeg opisany wieloma funkcjami

Na ogół modelowy przypadek geometrii obszaru z rys. 2, gdzie brzeg opisany jest przez jedną funkcję, jest rzadko spotykany. Brzeg obszaru jest opisany przez wiele funkcji, często nie dających się wyrazić jako funkcja tylko jednej zmiennej  $x$  lub  $y$ . Stopień skomplikowania zadania wzrasta, kiedy obszar nie jest jednoczynny bądź też składa się z rozłącznych podobszarów. Utrudnia to – a czasami uniemożliwia – algorytmizację, zadania dla takiego brzegu. Sposobem pozwalającym łatwo uwzględnić nawet najbardziej złożone przypadki geometrii obszaru jest wykorzystanie twierdzenia Greena.

## 4. Twierdzenie Greena

Równoważnym sposobem zastąpienia całek powierzchniowych w obszarze  $A$  przez odpowiednie wyrażenia na brzegu  $\Gamma$  jest wykorzystanie twierdzenia Greena (Stokesa):

$$\oint_{\Gamma} Pdx + Qdy = \iint_A \left( \frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right) dx dy \quad (24)$$

Twierdzenie zachodzi jedynie dla odpowiednio regularnych funkcji. Funkcje  $P(x,y)$  i  $Q(x,y)$  muszą być co najmniej klasy  $C^1$  wewnątrz obszaru  $A$ .

Najbardziej interesujące są dwa wzory opisujące przypadki szczególne (24), używane przy dowodzeniu.

$$\oint_{\Gamma} Pdx = - \iint_A \frac{\partial P}{\partial y} dx dy \quad (25)$$

$$\oint_{\Gamma} Qdy = \iint_A \frac{\partial Q}{\partial x} dx dy \quad (26)$$

Chcąc obliczyć wartość całki powierzchniowej  $C_{n,m}$ , wg równania (5), z wielomianu  $x^n y^m$ , można dobrać funkcję  $Q$  np. jako  $Q = x^{n+1} y^m$ , dla której zachodzi:

$$\iint_A x^n y^m dx dy = \frac{1}{n+1} \iint_A \frac{\partial(x^{n+1} y^m)}{\partial x} dx dy = \frac{1}{n+1} \oint_{\Gamma} x^{n+1} y^m dy \quad (27)$$

Analogicznie dla  $P = x^n y^{m+1}$  zachodzi:

$$\iint_A x^n y^m dx dy = \frac{1}{m+1} \iint_A \frac{\partial(x^n y^{m+1})}{\partial y} dx dy = - \frac{1}{m+1} \oint_{\Gamma} x^n y^{m+1} dx \quad (28)$$

Z tych równań wynikają, z dokładnością do znaku, wszystkie poprzednio pokazane wzory (6)–(11).

#### 4.1. Konsekwencje twierdzenia

Twierdzenie Greena prowadzi do niemal identycznych wzorów jak elementarne wyprowadzenie. Czysto matematyczne ujęcie problemu pokazuje nowe aspekty i w tym kontekście rozważyć należy: a) ciągłość funkcji podcałkowej, b) brzeg i jego opis.

##### 4.1.1. Ciągłość

Funkcje  $P$  i  $Q$  muszą być klasy  $C^1$ . Nie ma to znaczenia dla charakterystyk całego przekroju, gdyż funkcja podcałkowa – wielomian  $c_{n,m}$  definiowany przez (5) – jest klasy  $C^\infty$ . Jeśli oblicza się charakterystyki jedynie obszaru ściskanego, to funkcję podcałkową w całym obszarze przekroju należałoby przededefiniować jako:

$$c_{n,m}(x, y) = \begin{cases} x^n y^m & \text{ściskanie} \\ 0 & \text{rozciąganie} \end{cases} \quad (29)$$

Funkcja doznaje skoku na osi obrotu, przy przejściu ze ściskania do rozciągania. Wtedy w całym obszarze całkowania funkcja może nie być nawet klasy  $C^0$  i nie można wprost zastosować twierdzenia Greena. Obszar przekroju musi być podzielony na podobszary, w których związki konstytutywne są klasy  $C^1$ .

W przypadku NL istnieje jeden punkt nieciągłości krzywej konstytutywnej, między ściskaniem i rozciąganiem, więc należy podzielić przekrój na część ściskaną i rozciąganą. W przypadku innych związków konstytutywnych może być więcej punktów nieciągłości i wtedy obszar przekroju trzeba podzielić na podobszary, w których funkcja  $\sigma(\varepsilon)$  jest dostatecznie gładka. Ogólnie podział na podobszary odbywa się przez znalezienie prostych dla których  $\varepsilon(x, y) = \varepsilon_i^h$ , dla każdej charakterystycznej wartości  $\varepsilon_i^h$ , gdzie wykres  $\sigma(\varepsilon)$  ma nieciągłość funkcji lub pierwszej pochodnej.

Problem ciągłości może być jeszcze istotniejszy w przypadku obliczania (całkowania) macierzy stycznej. Wtedy w funkcji podcałkowej występuje nie  $\sigma(\varepsilon)$  ale  $\sigma'(\varepsilon)$ . Wymaga to podziału obszaru na fragmenty o ciągłości naprężeń  $C^2$ .

Podział przekroju na strefy jest zależny od punktów nieciągłości związków konstytutywnych. Z tego wynika też istotna wada wykorzystywania geometrii, gdyż trudno jest rozdzielić związki konstytutywne od sposobu całkowania ich w przekroju.

##### 4.1.2. Brzeg i jego opis

Poprzednie wzory (6)–(11) dotyczyły całki z funkcji, która jest tylko fragmentem brzegu, pozostała część brzegu była uwzględniana niejawnie przez boki bądź podstawę trapezu, gdyż dawały one zerowe całki. Wymagało to z kolei odpowiedniego podziału brzegu.

Twierdzenie Greena wymaga obliczenia całki po krzywej zamkniętej, czyli po wszystkich fragmentach brzegu. W konsekwencji całkę po konturze zamkniętej należałoby rozumieć jako sumę całek  $\oint_{\Gamma} = \sum_i \int_{\Delta\Gamma_i}$  po fragmentach brzegu  $\Delta\Gamma_i$ . W zamian nie ma potrzeby dokonywania jakiegokolwiek podziału. Ułatwia to algorytmizację, gdyż podział brzegu jest bardziej skomplikowany niż jednolite potraktowanie wszystkich fragmentów.

W ujęciu twierdzenia Greena brzeg jest krzywą zorientowaną. Orientacja krzywej na płaszczyźnie jest przyjmowana jako obrót od osi  $x$  do  $y$ . Skutkuje to odmiennym znakiem niż we wzorach (6)–(11), gdzie milcząco założono, że  $a < b$ . W przypadku użycia krzywej zorientowanej, brzeg obszaru z rys. 2 powinien być od  $b$  do  $a$ .

W ramach twierdzenia Greena przestaje się postrzegać brzeg jako funkcję  $x$  albo funkcję  $y$ . Jest tylko brzeg w pewnej parametryzacji. Jest to nieco mylące dla obszarów ograniczonych łamanymi, gdyż dla prostej współrzędne  $x$  bądź  $y$  są najbardziej naturalną parametryzacją. Przeszają one jednak odgrywać rolę argumentu i wartości funkcji.

Jednolite traktowanie wszystkich fragmentów brzegu jest największą praktyczną korzyścią, jaką wnosi twierdzenie Greena. Jak pokazano, wzory redukujące całą powierzchnię, do liniowej można otrzymać na innej drodze. Natomiast takie traktowanie brzegu umożliwiła algorytmizację obliczeń dla dowolnie skomplikowanego przekroju. To zostanie pokazane w pkt. 5.

Dlatego taki opis brzegu wart jest wykorzystania również wtedy, gdy nie korzysta się z twierdzenia Greena. W szczególności pozwala on uogólnić podejścia używane przez Boneta [1, 2], Dundara [7] czy Zupana [16] na dowolnie złożone przekroje.

#### 4.2. Zestaw wzorów

Wzory wynikające z twierdzenia Greena umożliwiają obliczanie charakterystyk na wiele, równoważnych sposobów. Wybór jest dosyć arbitralny. Poniżej zestawiono te używane w implementacji. Do oznaczeń obliczanej wielkości dodano symbol  $\Delta$ , gdyż są to jedynie przyrosty, które w odróżnieniu od (6)–(11) nigdy nie występują samodzielnie.

$$\Delta A = y_p(x_a - x_b) \quad (30)$$

$$\Delta S_x = \frac{1}{6}(y_a^2 + y_a y_b + y_b^2)(x_a - x_b) \quad (31)$$

$$\Delta S_y = \frac{1}{12}(x_a^2 + 4x_p^2 + x_b^2)(y_b - y_a) \quad (32)$$

$$\Delta J_x = \frac{1}{18}(y_a^3 + 4y_p^3 + y_b^3)(x_a - x_b) \quad (33)$$

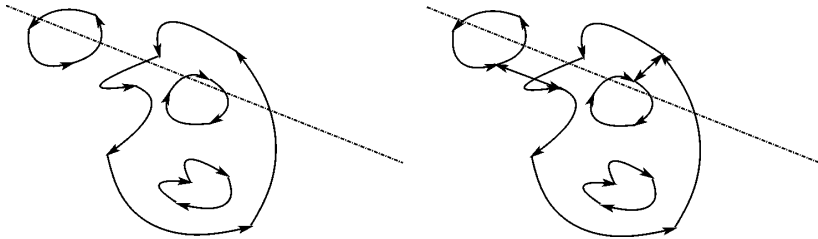
$$\Delta J_y = \frac{1}{18}(x_a^3 + 4x_p^3 + x_b^3)(y_b - y_a) \quad (34)$$

$$\Delta D_{xy} = \frac{1}{12}(x_a \cdot y_a^2 + 4x_p \cdot y_p^2 + x_b \cdot y_b^2)(x_a - x_b) \quad (35)$$

### 5. Reprezentacja brzegu

Jeśli obszar przekroju jest prostą geometrycznie figurą, nie widać wyraźnie zalet wykorzystania twierdzenia Greena w porównaniu do elementarnych wzorów. Występujące w praktyce kształty przekroju bywają znacznie bardziej skomplikowane. Dla obszaru nie-

wypukłego pojawiają się problemy z automatycznym podziałem na fragmenty. Dla obszaru niejednospójnego – z otworami w środku, por. rys. 3 (z lewej), pojawia się dodatkowy brzeg (bądź brzegi) wewnętrzny, analogiczny jak zewnętrzny, tyle że o przeciwnej orientacji.



Rys. 3. Warianty opisu obszaru wielospójnego: z wieloma brzegami bądź z jednym ciągłym brzegiem uzyskanym poprzez fikcyjne krawędzie (podwójne strzałki)

Fig. 3. Variants of multi-connected domain description: with multiple boundaries and with one boundary obtained by artificial boundaries (double arrows)

Możliwy jest również przekrój składający się z rozłącznych części, gdzie każda ma osobny brzeg zewnętrzny. Jest to o tyle istotny przypadek, że jeśli przekrój nie jest wypukły, wtedy część ściskana takiego przekroju może składać się z rozłącznych części.

Zalety twierdzenia Greena ujawniają się właśnie wtedy, kiedy przez odpowiedni opis brzegu można jednolicie opisać wszystkie te skomplikowane przypadki.

### 5.1. Opis obszaru i jego brzegu

Możliwe są dwa podejścia. Jeden opisany przez [14] (a także stosowany przez Palucha), gdzie przechowuje się osobne listy brzegów, każda zawierająca liczbę punktów. Jeden zewnętrzny – bądź więcej, jeśli obszar składa się z kilku rozłącznych podobszarów i kolejne brzegi wewnętrzne (o przeciwnym obiegu), o ile występują. Taka reprezentacja jest pokazana na rys. 3 (z lewej). W praktyce bywa to niedogodne, gdyż wymaga użycia bardziej złożonych struktur danych. Dlatego tutaj preferowany jest inny sposób.

Druga możliwość to wprowadzenie sztucznego brzegu między brzegami zewnętrznym a wewnętrznym, w postaci pary odcinków, od  $A$  do  $B$  i od  $B$  do  $A$ , przez co formalnie istnieje jeden ciągły brzeg – rys. 3 (z prawej). Brzeg fikcyjny nie zmienia wartości całki, gdyż wzdłuż pary krawędzi brzegu fikcyjnego funkcja podcałkowa jest ta sama, ale przeciwne są zwroty przyrostów  $dx$  lub  $dy$ . W analogiczny sposób można wprowadzić brzeg fikcyjny między rozłącznymi częściami obszaru. Stosując brzeg fikcyjny, można uzyskać jedną ciągłą linię brzegu we wszystkich możliwych przypadkach. Dla algorytmów operujących na przekroju najbardziej nawet złożony obszar topologicznie nie różni się od wypukłego wieloboku. Dotyczy to zarówno operacji sumowania (obliczania całek), jak i operacji znajdowania części ściskanej. Bardzo ułatwia to operacje znalezienia części ściskanej, gdyż wszystkie operacje przeprowadza się tak, jak na jednym wieloboku wypukłym.

## 5.2. Reprezentacja danych dla jednolitego przekroju

Przekrój jest reprezentowany przez wielobok opisujący przekrój. Ilość wierzchołków krzywej reprezentujących brzeg wieloboku jest zadana i nie zmienia się w trakcie obliczeń. W takiej sytuacji najbardziej neutralną reprezentacją w programie są macierze i wektory.

W algorytmie dane są reprezentowane przez tablicę (macierz)  $xy$ , w której wierszach są wpisane kolejne współrzędne  $x_i$  i  $y_i$  punktów brzegu. W tekście artykułu, z uwagi na zwartość zapisu wierszami w tekście, użyto osobnych wektorów dla  $x$  i  $y$  zamiast odpowiednich kolumn macierzy  $xy$ .

Oprócz tablicy  $xy$  używany jest wektor  $brzeg$ , w którym są zapisane numery kolejnych punktów tworzących brzeg. Przy takiej reprezentacji kolejne odcinki brzegu są opisane przez numery kolejnych par punktów z wektora  $brzeg$ . Ostatni odcinek od  $P_n$  do  $P_1$  trzeba uwzględnić dodatkowo.

W przypadku prostego geometrycznie kształtu przekroju dodatkowy wektor  $brzeg$  jest zbędny. Jeśli w tablicy  $xy$  współrzędne wierzchołków wieloboku są wpisane w kolejności obiegu oraz każdy wierzchołek występuje jeden raz, to odpowiada to wartości tablicy  $brzeg$  w postaci kolejnych liczb naturalnych, czyli identycznej jak numery kolejnych wierszy macierzy współrzędnych.

Jeśli zostanie wprowadzony brzeg fikcyjny, to niektóre punkty występują dwa lub więcej razy. Jest więc racjonalne, aby oddzielić położenie punktu od jego występowania na brzegu. Również uniemożliwia to zmianę współrzędnych punktu występującego dwukrotnie, aby jeden punkt mógł mieć różne współrzędne.

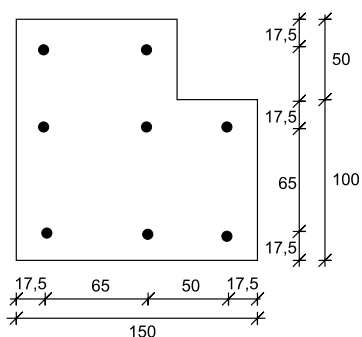
Co ważniejsze, w przypadku charakterystyk strefy ściskanej taka organizacja danych daje wielką elastyczność, gdyż można pominąć wierzchołki rozciągane, nie usuwając ich z tablicy. Zostanie to wykorzystane przy obliczeniach dla strefy ściskanej i pokazane w pkt. 6.

Dla przekroju pokazanego na rys. 4, zaczerpniętym z [7] oraz analizowanym przez autora [10], dane geometryczne w układzie współrzędnych w dolnym lewym rogu przedstawiają się następująco (dolne indeksy oznaczające pozycję na liście dodane są dla czytelności):

$$x = \{0.0_1, 150_2, 150_3, 100_4, 100_5, 0.0_6\}$$

$$y = \{0.0_1, 0.0_2, 100_3, 100_4, 150_5, 150_6\}$$

$$brzeg = \{1_1, 2_2, 3_3, 4_4, 5_5, 6_6\}$$



Rys. 4. Analizowany przekrój (wymiary w mm)

Fig. 4. Analysed example of cross-section (dimensions in mm)

### 5.3. Reprezentacja dla przekroju złożonego

Dla przekroju złożonego z kilku materiałów na ogół potrzeba obliczyć charakterystyki dla każdego materiału osobno, gdyż w dalszych obliczeniach te charakterystyki będą uwzględniane z jakąś wagą (typowo moduł Younga, por. (4)).

Przekrój złożony byłby opisywany poprzez zestaw wieloboków z których każdy opisuje część przekroju z jednego materiału. Wieloboki te nie powinny mieć części wspólnej, ewentualnie za wyjątkiem brzegu fikcyjnego.

Typowym przykładem może być przekrój zespolony stalowo-betonowy, gdzie występowałyby wielobok opisujący część betonową oraz drugi wielobok opisujący geometrię profili stalowych. Analogicznie należałoby traktować przekrój żelbetowy ze zbrojeniem sztywnym.

#### 5.3.1. Przekrój żelbetowy

Opis przekroju w postaci dwu wieloboków definiujących osobno kształt betonu i stali nie jest dobrą strategią w przypadku przekroju żelbetowego składającego się z betonu oraz prętów zbrojenia miękkiego. Zbrojenie występuje w postaci licznych prętów o niewielkich rozmiarach, więc opisanie każdego pręta łamaną wymagałoby użycia dużej liczby odcinków łamanej i długiego czasu obliczeń. Ponieważ pręty są z natury okrągłe, zbędny jest jawny opis kształtu. Bardziej efektywna jest reprezentacja przez wielobok opisujący cały przekrój oraz dane opisujące zbrojenie. Wystarczy tablica współrzędnych prętów oraz tablica średnic prętów.

Dla takiej reprezentacji przekroju żelbetowego potraktowanie wieloboku przekroju jako wieloboku definiującego beton daje zawyżone wartości charakterystyk, tzw. wartości brutto, gdyż powierzchnię rzeczywiście zajmowaną przez zbrojenie wlicza się do powierzchni betonu.

Różnice charakterystyk brutto i netto są zazwyczaj mało istotne dla całości przekroju, gdyż stal zajmuje niewielką część całego przekroju. Natomiast w przypadku obliczania charakterystyk dla części ściskanej przekroju te proporcje mogą się istotnie zmienić i różnica może być znacząca.

Istnieją trzy możliwości potraktowania zagadnienia. Pierwsza polega na ignorowaniu problemu. Charalampakis [4] krytykuje praktykę traktowania zbrojenia jako punktów pozabawionych powierzchni – por. Cedolin [3], Sousa [14] – co odpowiada uwzględnianiu zbrojenia w części ściskanej z modułem Younga  $(\alpha + 1)E^b$ , gdzie  $\alpha = \frac{E^s}{E^b}$ .

Druga wywodzi się z koncepcji analitycznej, gdy w strefie ściskanej zbrojenie uwzględnia się z modułem Younga  $(\alpha - 1)E^b$  a w rozciąganej z mnożnikiem  $\alpha$ .

Trzecia polega na rozdzieleniu obliczeń dla obu materiałów. Oblicza się charakterystyki brutto dla betonu i odejmuje charakterystyki betonu dla kół zajmowanych przez zbrojenie. Charakterystyki dla stali oblicza się bez żadnych mnożników.

Wyniki są równoważne drugiej koncepcji, ale z punktu widzenia programowania jest to lepsza strategia. Charakterystyki dla stali i betonu stają się niezależne, więc łatwo poddać je modyfikacji.

#### 5.4. Algorytm charakterystyk

Algorytm obliczania charakterystyk podany na rys. 5 polega na obliczaniu przyrostów wg wzorów (30)–(35) dla kolejnych odcinków tworzących brzeg obszaru.

Tablica `brzeg` zawiera kolejne punkty brzegu, więc para kolejnych punktów definiuje odcinek. Osobnego uwzględnienia wymaga jedynie ostatni zamykający odcinek od  $P_n$  do  $P_1$ . To jest realizowane w instrukcji z adnotacją „cyklicznie”, gdzie po osiągnięciu końca wektora wraca się na pozycję pierwszą.

Dla znanych numerów punktów początkowego  $a$  i końcowego  $b$  dla danego odcinka z tablicy `xy` pobierane są współrzędne początku  $x_a$  i  $y_a$  odcinka, końca  $x_b$  i  $y_b$  oraz obliczane współrzędne punktu w środku  $x_p$  i  $y_p$ .

Dla tych wszystkich współrzędnych obliczane są przyrosty według wzorów (30)–(35). Przyrosty te dodawane są do odpowiednich wielkości całkowitych.

```

Wyzeruj charakterystyki
for  $i = 1$  to rozmiar tablicy brzeg do
   $a = \text{brzeg}(i)$ 
   $b = \text{brzeg}(i+1)$  (cyklicznie)
   $x_a = xy(a, 1)$    $y_a = xy(a, 2)$ 
   $x_b = xy(b, 1)$    $y_b = xy(b, 2)$ 
  oblicz  $x_p, y_p$ 
  oblicz przyrosty wg. wzorów (30)–(35)
  dodaj przyrosty do wielkości całkowitych
end for
{Tylko w przypadku strefy ściskanej}
if  $A < 0$  then
  Wyzeruj charakterystyki – por. pkt. 6.3
end if

```

Rys. 5. Algorytm obliczania charakterystyk geometrycznych dla wieloboku zadanego przez ciąg punktów

Fig. 5. Polygon moment of area computing algorithm

Obliczanie charakterystyk strefy ściskanej formalnie nie różni się niczym, jeśli zamiast wieloboku, opisującego cały przekrój przekazany zostanie do obliczeń wielobok opisujący strefę ściskaną przekroju. W rzeczywistości dane muszą być odpowiednio zorganizowane. Specyfikę opisu zbrojenia już poruszono. Inne zależności wynikające z użycia ogólnego algorytmu obliczania charakterystyk do części ściskanej przekroju zostaną podane później w punkcie 6.3.

### 6. Algorytm wyznaczania strefy ściskanej

Przedstawiony zostanie tutaj najprostszy koncepcyjnie – niekoniecznie najbardziej wydajny – algorytm polegający na trzech przebiegach listy punktów brzegu. Jest on podobny do [14], gdzie jednak został bardzo lakonicznie potraktowany. Tutaj zostanie opisany na tyle szczegółowo, aby można było go zaimplementować.

Zadanie wyznaczenia strefy ściskanej jest zadaniem czysto geometrycznym. Obszar (tutaj wielobok) przekroju jest w części ściskany, a w części rozciągany. Wielobok opisujący obszar rozciągany jest w tym przypadku nieistotny, więc zadanie sprowadza się do znalezienia jedynie jednego wieloboku – opisującego część ściskaną.

Granicą między częścią ściskaną a rozciąganą jest oś obrotu, gdzie odkształcenia są zerowe. Równanie osi obrotu można uzyskać z (1) i opisuje ono prostą.

Równoważnie można więc sformułować to zadanie: wyznaczyć nowy wielobok leżący po zadanej stronie prostej.

Poszukiwany wielobok będzie składał się z (części) punktów wieloboku przekroju i punktów przecięcia brzegu z osią obrotu. Znalezienie punktów wieloboku w przypadku obliczeń analitycznych nie sprawia żadnych kłopotów, widoczne są relacje geometryczne i topologiczne. Z punktu widzenia algorytmu realizowanego przez komputer zadanie jest o wiele trudniejsze.

Analogiczny problem występuje w geometrii obliczeniowej, gdzie do badania po której stronie prostej leży dany punkt używa się iloczynu wektorowego. Jeśli prosta jest zadana przez dwa (uporządkowane) punkty  $A$  i  $B$  i należy sprawdzić po której stronie znajduje się punkt  $C$ , to zwrot wektora  $\vec{AB} \times \vec{AC}$  określa położenie. Jest to prosty test ale niestety zawodny.

Jeśli punkt  $B$  leży dostatecznie blisko prostej, to z powodu skończonej dokładności arytmetyki procesora, wartość jest zerowa. Wtedy nie można stwierdzić, po której stronie prostej leży zadany punkt. Prowadzi to do bardzo skomplikowanych rozwiązań, typu arytmetyka adaptacyjna zaproponowana przez Shewchuka [13].

W przypadku wyznaczania strefy ściskanej można częściowo uniknąć problemu, korzystając z wartości odkształceń. Płaszczyzna odkształceń (1) przecina płaszczyznę przekroju na osi obrotu. Odkształcenia są ujemne w części ściskanej, a dodatnie w części rozciąganej. Zamiast badać położenie punktu, wygodniej sprawdzić znak odkształceń.

Nie jest to recepta na arytmetykę procesora. Raczej skutki błędów zaokrągleń są inne. W geometrii obliczeniowej jeśli punkt (pixel), który jest zakryty płaszczyzną (leży wewnątrz zakrytego obszaru) pojawi się na rysunku, to jest to widoczna wada. Jeśli punkt obszaru ściskanego znajdzie się, w wyniku błędów zaokrągleń, w obszarze rozciągany, to wpływ tego niewielkiego przesunięcia na charakterystyki geometryczne jest niezauważalny. Nie oznacza to, że problem przestaje istnieć. Dalej, przy poszczególnych krokach algorytmu, podano komentarz dotyczący tego, jakie skutki może mieć to zjawisko.

Algorytm wyznaczania strefy ściskanej nie musi być wydajny obliczeniowo. Jego złożoność wynosi i tak  $O(n)$ , a liczba punktów jest niewielka – typowy przekrój ma kilka, kilkanaście punktów. Nawet bardzo dokładnie zdyskretyzowany brzeg krzywoliniowy będzie miał rzędu tysiąca punktów. Dopiero przy wielokrotnym poszukiwaniu strefy ściskanej, np. przy wyznaczaniu obwiedni nośności przekroju wydajność algorytmu może mieć znaczenie.

Opisując algorytm, będzie jednocześnie pokazywane jego działanie na przykładzie podanym na rys. 4. Aby zilustrować działanie, potrzeba wcześniej ustalić reprezentację danych.



## 6.1. Reprezentacja danych w programie

W przypadku wyznaczania strefy ściskanej ilość wierzchołków wieloboku opisującego strefę ściskaną nie jest znana. Może zarówno maleć, jak i rosnać, w stosunku do ilości wierzchołków całego wieloboku.

Wymaga to użycia jakiejś formy dynamicznych struktur danych. Obliczenia nie są skomplikowane, najistotniejsze okazują się być operacje na strukturach danych. Algorytm w dużym stopniu jest przez nie określony.

Naturalną reprezentacją stają się w tym przypadku listy. Jednak użycie list nie jest zbyt łatwą techniką, nie wszystkie języki programowania posiadają taką możliwość. Istnieje długa tradycja używania w tym celu macierzy o rozmiarze większym niż początkowe dane. To daje możliwość dopisywania dodatkowych danych na końcu tablicy. Ceną, jaką płaci się za to uproszczenie, jest konieczność zabezpieczenia się przed przekroczeniem końca tablicy oraz problem wyboru, ile dodatkowego miejsca zostanie zarezerwowane.

Jeszcze inną możliwością jest wykorzystanie tablic o zmiennym rozmiarze, które w razie potrzeby powiększają swój rozmiar. Niektóre języki udostępniają takie typy danych. W tej konwencji będą formułowane dalsze algorytmy. Uzupełnienie algorytmu o deklarację rozmiaru i zabezpieczenie przed przekroczeniem jest rutynowym zadaniem i zostanie tutaj pominięte.

Dane w postaci tablic `xy` i `brzeg` zawierają te same wartości co w punkcie 5.2, ale teraz reprezentacja jest nieco inna. O ile wektor `brzeg` opisujący wierzchołki całego przekroju ma ustalony i niezmienny rozmiar, o tyle tablica `xy` musi umożliwiać dopisywanie nowych punktów, więc wymaga stosownej reprezentacji.

Niezbędne są też zadane parametry określające oś obrotu (1). W przykładzie przyjęto:  $\varepsilon_0 = 10^{-3}$ ,  $\kappa_y = -3.0137 \cdot 10^{-3} \frac{1}{m}$  i  $\kappa_x = -6.849 \cdot 10^{-3} \frac{1}{m}$ .

Wierzchołki wieloboku opisującego część ściskaną będą zapisywane najpierw do wektora tymczasowego `tmp` a następnie do nowego wektora `nowy_brzeg`. Ich rozmiar nie jest znany na początku.

Potrzebny będzie też wektor `eps` do przechowywania wartości odkształceń we wszystkich wyznaczonych punktach. Jego rozmiar jest identyczny jak rozmiar `xy`, włącznie z ilością dostępnego miejsca.

## 6.2. Opis algorytmu

**Przebieg 1.** Pierwszy przebieg ogranicza się do obliczenia odkształceń w każdym punkcie występującym w tablicy `xy`. Obliczona wartość  $\varepsilon$  jest wpisywana na odpowiednią pozycję wektora `eps`. Z punktu widzenia opisu algorytmu istotny jest znak, więc poniżej wpisano jedynie znaki odkształceń.

$$\text{eps} = \{ (+)_1 \quad (+)_2 \quad (-)_3 \quad (+)_4 \quad (-)_5 \quad (-)_6 \}$$

W zasadzie niemal niemożliwa jest sytuacja, że natrafia się na wartość zerową. Jest to bardzo rzadkie (reprezentacja wartości 0.0 jest specjalna), jednak nie można tego wykluczyć.

**Przebieg 2.** W drugim przebiegu punkty brzegu z tablicy  $\text{brzeg}$  są przepisywane do tablicy  $\text{tmp}$  oraz uzupełniane – w kolejności występowania – o punkty, gdzie oś obrotu przekroju przecina brzeg.

Brane są pary kolejnych punktów (czyli odcinek) z wektora  $\text{brzeg}$  i badany jest znak  $\varepsilon_a \cdot \varepsilon_b$ . Ujemny – dokładnie ujemny a nie  $\leq 0$ , teoretycznie możliwe jest, że w punkcie jest zerowe odkształcenie – świadczy o istnieniu punktu pośredniego, o współrzędnych  $\bar{x}, \bar{y}$  gdzie  $\varepsilon = 0$ . Należy wyznaczyć położenie tego punktu i wpisać na listę między końce odcinka.

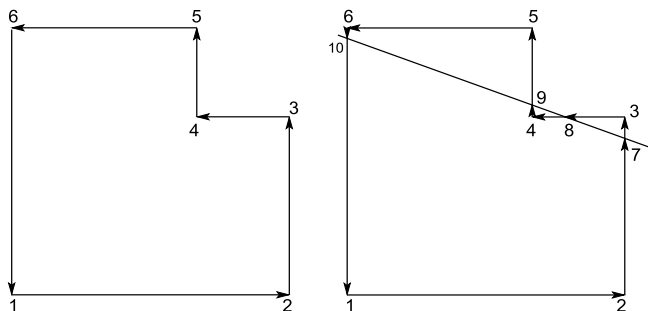
Przy wyznaczaniu położenia punktu pośredniego można wykorzystać zależność, że wszystkie wielkości zmieniają się liniowo, więc zachodzi dla  $t \in [0, 1]$ .

$$\begin{cases} x(t) = x_a + t(x_b - x_a) \\ y(t) = y_a + t(y_b - y_a) \\ \varepsilon(t) = \varepsilon_a + t(\varepsilon_b - \varepsilon_a) \end{cases} \quad (36)$$

Z trzeciego równania można wyznaczyć  $\bar{t}$  dla którego  $\varepsilon = 0$

$$\bar{t} = \frac{\varepsilon_a}{\varepsilon_a - \varepsilon_b}$$

Znając  $\bar{t}$ , można obliczyć współrzędne punktu pośredniego,  $\bar{x} = x(\bar{t})$ ,  $\bar{y} = y(\bar{t})$ , korzystając z dwu pierwszych równań (36).



Rys. 6. Wielobok przekroju oraz przekrój z osią obrotu i wyznaczonymi punktami pośrednimi

Fig. 6. Cross-section polygon and polygon with axis of rotation and with midpoint

Do tablicy współrzędnych  $x\bar{y}$  dopisywany jest kolejny punkt, do tablicy nowego konturu  $\text{tmp}$  wpisywany jest poprzedni punkt oraz wyznaczony punkt pośredni. Do tablicy odkształceń  $\varepsilon\bar{p}$ , wpisywana jest zerowa wartość odkształcenia, przy czym jest istotne, aby wpisywać wartość 0.0, a nie wartość  $\varepsilon$  obliczoną dla  $\bar{x}$  i  $\bar{y}$ . Ta ostatnia może z powodu błędów zaokrągleń być nieco różna od zera (zero numeryczne) natomiast wartość zerowa jest używana jako charakterystyczna w dalszej części algorytmu.

Tutaj ujawnia się znowu problem skończonej dokładności arytmetyki. Jeśli w pewnym punkcie  $P_k$  wartość odkształceń jest prawie zero (zero numeryczne), to współrzędne  $\bar{x}$  i  $\bar{y}$  mogą numerycznie nie różnić się od  $x_k$  i  $y_k$ , ale formalnie pojawi się na brzegu nowy punkt, który geometrycznie pokrywa się z już istniejącym.

Pojawienie się podwójnego punktu na brzegu, z uwagi na budowę wzorów (30)–(35) nie wpłynie w żaden sposób na charakterystyki, gdyż wszędzie występuje mnożenie przez długość. Możliwość pojawienia się podwójnego punktu nie przeszkadza na etapie obliczeń, natomiast stwarza problemy przy próbie powrotu do reprezentacji bez brzegu fikcyjnego (wykonywania rysunków).

Po zrealizowaniu drugiego kroku algorytmu, jak to zostało pokazane na rys. 6 (z prawej), dane rozwiązane przykładowo przedstawiają się tak, że tablica  $xy$  zawiera punkty:

$$x = \{ 0.0_1 \quad 150_2 \quad 150_3 \quad 100_4 \quad 100_5 \quad 0.0_6 \quad 150_7 \quad 104.54_8 \quad 100_9 \quad 0.0_{10} \}$$

$$y = \{ 0.0_1 \quad 0.0_2 \quad 100_3 \quad 100_4 \quad 150_5 \quad 150_6 \quad 80_7 \quad 100_8 \quad 102_9 \quad 146_{10} \}$$

Po modyfikacji tablica odkształceń zawiera następujące wartości:

$$\text{eps} = \{ (+)_1 \quad (+)_2 \quad (-)_3 \quad (+)_4 \quad (-)_5 \quad (-)_6 \quad (=)_7 \quad (=)_8 \quad (=)_9 \quad (=)_{10} \}$$

Tablica  $\text{tmp}$  zawiera następujące punkty:

$$\text{tmp} = \{ 1_1 \quad 2_2 \quad 7_3 \quad 3_4 \quad 8_5 \quad 4_6 \quad 9_7 \quad 5_8 \quad 6_9 \quad 10_{10} \}$$

```

{Krok 1}
for  $i = 1$  to ilość wierszy tablicy  $xy$  do
    oblicz  $\varepsilon$  z równania (1)
     $\text{eps}(i) = \varepsilon$ 
end for
{Krok 2}
for  $i = 1$  to rozmiar tablicy  $\text{brzeg}$  do
     $a = \text{brzeg}(i)$ 
     $b = \text{brzeg}(i+1)$  (cyklicznie)
    wpisz  $a$  na koniec tablicy  $\text{tmp}$ 
    if  $\varepsilon_a \times \varepsilon_b < 0$  then
        oblicz  $\bar{x}, \bar{y}$ 
        wpisz punkt pośredni  $\bar{x}, \bar{y}$  na koniec tablicy  $xy$ , pozycja  $p$ 
        wpisz  $p$  na koniec tablicy  $\text{tmp}$ 
        wpisz  $\text{eps}(p) = 0.0$ 
    end if
end for
{Krok 3}
 $c = 1$ ;
for  $a = 1$  to rozmiar tablicy  $\text{tmp}$  do
    if  $\text{eps}(a) \leq 0$  then
         $\text{nowy\_brzeg}(c) = \text{tmp}(a)$ 
         $c = c + 1$ 
    end if
end for

```

Rys. 7. Algorytm wyznaczania strefy ściskanej

Fig. 7. Algorithm for finding the cross-section compressed region

**Przebieg 3.** W trzecim przebiegu przepisana zostaje lista punktów brzegu tmp do trzeciego wektora nowy\_brzeg, z pominięciem punktów, gdzie  $\varepsilon > 0$  (rozciąganie, zero to punkty graniczne), czyli przepisane są punkty dla których  $\varepsilon \leq 0$ .

Po wykonaniu trzeciego kroku algorytmu, co pokazano na rys. 8, tablica xy zawiera następujące punkty:

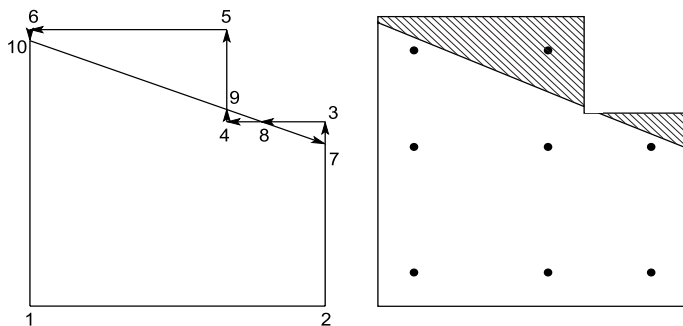
$$x = \{ 0.0_1 \quad 150_2 \quad 150_3 \quad 100_4 \quad 100_5 \quad 0.0_6 \quad 150_7 \quad 104.54_8 \quad 100_9 \quad 0.0_{10} \}$$

$$y = \{ 0.0_1 \quad 0.0_2 \quad 100_3 \quad 100_4 \quad 150_5 \quad 150_6 \quad 80_7 \quad 100_8 \quad 102_9 \quad 146_{10} \}$$

Tablica nowy\_brzeg zawiera następujące punkty:

$$\text{nowy\_brzeg} = \{ 7_1 \quad 3_2 \quad 8_3 \quad 4_4 \quad 9_5 \quad 5_6 \quad 6_7 \quad 10_8 \}$$

Warto zwrócić uwagę, że krawędź od  $P_{10}$  do  $P_7$  nie składa się z osobnych części  $P_{10}P_9$ ,  $P_9P_8$  i  $P_8P_7$ . Jest jednym odcinkiem.



Rys. 8. Wielobok wyznaczonego obszaru ściskanego oraz widok części ściskanej przekroju

Fig. 8. Determined polygon of compressed region and real compressed region of cross-section

Całość operacji, wszystkie przebiegi zestawiono na rys. 7. Po zrealizowaniu algorytmu w wektorze nowy\_brzeg są punkty brzegu obszaru ściskanego. Opis obszaru ściskanego jest w dokładnie takiej samej formie jak opis przekroju. Tablicę xy oraz wektor nowy\_brzeg można przekazać jako dane opisujące przekrój do algorytmu obliczającego charakterystyki geometryczne podanego na rys. 5.

Taka reprezentacja danych jest szczególnie efektywna w przypadku iteracji, czyli poszukiwania równowagi dla kolejnych przybliżeń osi obrotu, gdyż w tablicy xy są stałe wierzchołki całego przekroju. Dopisywanie odbywa się zawsze poza zakresem, gdzie znajdują się wierzchołki przekroju. Kolejne iteracje można wykonywać na tej samej tablicy, nadpisując punkty pośrednie.

Tablica xy jest jedna, ale wektor (lista) brzeg opisuje cały przekrój, a lista nowy\_brzeg część ściskaną.

### 6.3. Powiązanie obu algorytmów

Przy tak zorganizowanych danych wejściowych funkcja obliczająca charakterystyki (rys. 5) jest niezależna od innych części – można jej używać do obliczania dowolnego przekroju, w całości bądź w części. Jednak możliwość użycia jej do obliczania charakterystyk części ściskanej wymaga dostosowania do specyficznego kontekstu.

Dla charakterystyk całego przekroju pole powierzchni powinno być dodatnie. Wartość zerowa traktowana jest jako błąd danych, gdyż oznacza to brak przekroju. Dla charakterystyk części ściskanej cały przekrój może być rozciągany, a więc pole powierzchni części ściskanej może być zerowe, co jest poprawnym wynikiem.

Możliwe jest również, że ściskanie występuje jedynie w obszarach miary zero. Takie zdegenerowane obszary generują listę punktów brzegu składającą się z mniej niż trzech punktów. Należy zabezpieczyć się przed próbą obliczeń dla listy brzegu składającego się z mniej niż trzy punkty. W szczególności niebezpieczny jest przypadek pustej listy punktów – algorytm charakterystyk (rys. 5) nie załamuje się i daje poprawne wyniki już dla jednego punktu.

Nie wyczerpuje to listy możliwych problemów.

#### 6.3.1. Trójkąt o niewłaściwej orientacji. Błąd wyznaczenia strony

Możliwe jest nawet uzyskanie ujemnej wartości pola powierzchni. Jest to kolejny skutek skończonej precyzji arytmetyki procesora, kiedy punkt przecięcia zostanie wyznaczony po niewłaściwej stronie brzegu i w efekcie obszar będzie miał odwrotną orientację. Ponieważ efekt ten ujawnia się jedynie w przypadkach, kiedy obszar ściskany jest bardzo bliski zeru, należy to traktować jako brak przekroju ściskanego. W konsekwencji, procedurę obliczającą charakterystyki należałoby uzupełnić o warunek, że jeśli  $A < 0$  to wartości wszystkich charakterystyk są zerowe, co jest uwzględnione na rys. 5.

### 6.4. Uogólnienie

Algorytm wyznaczania strefy ściskanej daje się w sposób naturalny uogólnić na dowolne związki konstytutywne dane przedziałami. Wymagany jest wtedy podział przekroju na strefy, gdzie obowiązuje jeden z przedziałów związków konstytutywnych. Modyfikacji wymaga jedynie warunek na punkty pośrednie do postaci  $(\varepsilon_a - \varepsilon_i^h)(\varepsilon_b - \varepsilon_i^h) < 0$ , gdzie wielkości  $\varepsilon_i^h$  określają punkty charakterystyczne związków konstytutywnych.

W przypadku związków odcinkowo-liniowych poprzez charakterystyki można wyrazić wszystkie potrzebne wielkości przekrojowe. Jeśli związki nie są liniowe w danym przedziale, obliczanie charakterystyk na ogół nie wystarcza do rozwiązania zadania, jednak podział na strefy jest dalej użyteczny.

W przypadku podziału na strefy pojawia się dodatkowy problem; dla jednorodnego stanu odkształcenia będącego którąś z wartości  $\varepsilon_i^h$  otrzymuje się podwójne wyniki. Można tego uniknąć, znajdując w pierwszym przebiegu  $\varepsilon_{\min}$  i  $\varepsilon_{\max}$ . Jeśli się nie różnią, to jest to stan jednorodny.

## 7. Wnioski

Obliczanie charakterystyk strefy ściskanej przekroju jest efektywniejsze, kiedy zostanie rozdzielone na dwa etapy: wyznaczenie strefy ściskanej oraz obliczanie charakterystyk.

W artykule do obliczania charakterystyk wykorzystano całki brzegowe zamiast powierzchniowych, co bardzo upraszcza i przyspiesza obliczenia.

Wyprowadzono na drodze elementarnej zestaw wzorów służących od obliczania charakterystyk, dodatkowe uproszczenia w wyprowadzeniach były możliwe dzięki wykorzystaniu kwadratury Simpsona w nietypowej roli.

Pokazano, że równoważne wzory można otrzymać wykorzystując twierdzenie Greena. Jednak najważniejszą korzyścią, jakie wnosi to twierdzenie, jest sposób traktowania brzegu, dzięki któremu najbardziej nawet złożony przekrój topologicznie nie różni się od wypukłego wieloboku.

Szczegółowo potraktowano reprezentację danych niezbędnych do opisu brzegu przekroju. Zestawiono dokładny algorytm obliczania charakterystyk geometrycznych obszaru. Opisanano współpracujący algorytm wyznaczania strefy ściskanej przekroju. Poruszono problemy numeryczne występujące przy realizacji tego typu algorytmów.

Artykuł szeroko opisuje kwestie obliczania charakterystyk strefy ściskanej dla przekroju, którego brzeg daje się opisać łamanymi, jednak zdaniem Autora, brakuje algorytmu – wraz z reprezentacją danych – który uwzględniałby występowanie na brzegu kilku fragmentów łuku kołowego.

## Literatura

- [1] Bonet J.L., Barros M.H.F.M, Romero M.L., *Comparative study of analytical and numerical algorithms for designing reinforced concrete sections under biaxial bending*, Computers & Structures, 84(31–32), 2006, 2184-2193.
- [2] Bonet J.L., Romero M.L., Miguel P.F, Fernandez M.A., *A fast stress integration algorithm for reinforced concrete sections with axial loads and biaxial bending*, Computers & Structures, 82(2–3), 2004, 213-225.
- [3] Cedolin L., Cusatis G., Eccheli S., Roveda M., *Capacity of Rectangular Cross Sections under Biaxially Eccentric Loads*, ACI Structural Journal, 105(2), 2008, 3-4.
- [4] Charalampakis A.E., Koumousis V.K., *Ultimate strength analysis of composite sections under biaxial bending and axial load*, Advances in Engineering Software, 39(11), 2008, 923-936.
- [5] Dias da Silva V., Barros M.H.F.M, Júlio E.N.B.S., Ferreira C.C., *Closed form ultimate strength of multi-rectangle reinforced concrete sections under axial load and biaxial bending*, Computers and Concrete, 6(6), 2009, 505-521.
- [6] Di Ludovico M., Lignola G.P., Prota A., Cosenza E., *Nonlinear Analysis of Cross Sections under Axial Load and Biaxial Bending*, ACI Structural Journal, 107(4), 7/8, 2010, 390-399.
- [7] Dunder C., Tokgoz S., Tanrikulu A.K., Baran T., *Behaviour of reinforced and concrete-encased composite columns subjected to biaxial bending and axial load*, Building and Environment, 43(6), 2008, 1109-1120.
- [8] Izzuddin B.A., Siyam A.A.F.M., Lloyd-Smith D., *An efficient beam-column formulation for 3D reinforced concrete frames*, Computers & Structures, 80(7-8), 2002, 659-676.

- [9] Kawakami M., Ghali A., *Time-Dependent Stresses in Prestressed Concrete Sections of General Shape*, PCI Journal, 41, 5-6, 1996, 96-102.
- [10] Matuszak A., *Algorytm wyznaczania strefy ściskanej w zagadnieniu zakotwień oraz w przypadku przekroju żelbetowego projektowanego według NL*, Czasopismo Techniczne, 108(2-B), 2011, 49-78.
- [11] Paluch M., *Mechanika teoretyczna*, Wydawnictwo Politechniki Krakowskiej, Skrypt, 1988.
- [12] Sfakianakis M.G., *Biaxial bending with axial force of reinforced, composite and repaired concrete sections of arbitrary shape by fiber model and computer graphics*, Advances in Engineering Software, 33(4), 2002, 227-242.
- [13] Shewchuk J.R., *Robust Adaptive Floating-Point Geometric Predicates*, Proceedings of the Twelfth Annual Symposium on Computational Geometry, Association for Computing Machinery, maj 1996, 141-150.
- [14] Sousa J.B.M., Cereno F.D.G., Muniz, *Analytical integration of cross section properties for numerical analysis of reinforced concrete, steel and composite frames*, Engineering Structures, 29(4), 2007, 618-625.
- [15] Zak M.L., *Computer analysis of reinforced concrete sections under biaxial bending and longitudinal load*, ACI Structural Journal, 90(2), 3-4, 1993, 163-169.
- [16] Zupan D., Saje M., *Analytical integration of stress field and tangent material moduli over concrete cross-sections*, Computers & Structures, 83(28-30), 2005, 2368-2380.