

Zastosowanie pakietu $\mathcal{X}\mathcal{Y}$ -pic do konstrukcji diagramów macierzowych

Romuald Słupski

Uczelniane Centrum Informatyczne UMK
87-100 Toruń, ul. Chopina 12/18
rbs@uni.torun.pl

Pracę zgłosił: Andrzej Borzyszkowski

Streszczenie

Język diagramatyczny jest wygodnym narzędziem stosowanym w wielu dziedzinach nauki. Do narysowania diagramu można posłużyć się systemem $\text{T}\mathcal{E}\text{X}$ wspomaganym przez zbiory makrodefinicji, ułatwiających tworzenie obiektu graficznego. W artykule opisano generowanie diagramów macierzowych za pomocą pakietu $\mathcal{X}\mathcal{Y}$ -pic i podano konkretne przykłady zastosowań.

Wprowadzenie

Język diagramatyczny jest wygodnym narzędziem do przedstawiania równań, własności obiektów i procesów w wielu działach matematyki (teoria kategorii, teoria automatów) i fizyki teoretycznej (wielociałowy rachunek zaburzeń, teoria sprzężonych klasterów).

Do konstruowania diagramów można posłużyć się systemem $\text{T}\mathcal{E}\text{X}$. Istnieje wiele zbiorów makrodefinicji $\text{T}\mathcal{E}\text{X}$ -owych, wspomagających tworzenie diagramów. Są to np.:

- **pb-diagram** – zbiór makrodefinicji Paula Burcharda służący do tworzenia diagramów komutatywnych (*commutative diagram*) [1],
- **feynMF** – zbiór, napisanych przez Thorstena Ohl'a, makrodefinicji służących do rysowania diagramów Feynmana, w którym wykorzystuje się Metafont-a lub Metapost-a [2],
- **diagrams** – pakiet makrodefinicji Paula Taylora [3],
- **DCpic** – zestaw makrodefinicji Pedro Quaresma de Almeida (w języku portugalskim: DC – Diagramas Comutativos) [4],
- **$\mathcal{X}\mathcal{Y}$ -pic** – pakiet, którego autorami są Kristoffer H. Rose i Ross Moore [6, 7, 8],
- **diagxy** – pakiet Michaela Barra, wykorzystujący $\mathcal{X}\mathcal{Y}$ -pic [5].

Niniejszy artykuł poświęcony jest zastosowaniu pakietu $\mathcal{X}\mathcal{Y}$ -pic do konstruowania diagramów.

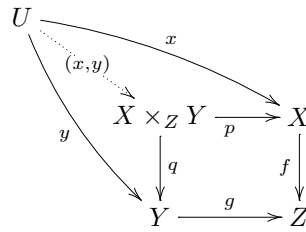
Pakiet $\mathcal{X}\mathcal{Y}$ -pic

$\mathcal{X}\mathcal{Y}$ -pic jest pakietem do składania grafów i diagramów za pomocą systemu $\text{T}\mathcal{E}\text{X}$. Współpracuje z wieloma dostępnymi formatami, np.: plain $\text{T}\mathcal{E}\text{X}$, $\text{L}\mathcal{A}\text{T}\mathcal{E}\text{X}$, $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\text{L}\mathcal{A}\text{T}\mathcal{E}\text{X}$ i $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\text{T}\mathcal{E}\text{X}$. Pakiet $\mathcal{X}\mathcal{Y}$ -pic definiuje dogodny język notacji, który służy do opisu struktur i obiektów graficznych. Sposób konstrukcji diagramu oparty jest na zasadzie logicznego opisu wizualnych składników grafu (*the logical composition of visual components*). Pozwala to na intuicyjne budowanie, niekiedy złożonej, struktury obiektu graficznego. Pakiet $\mathcal{X}\mathcal{Y}$ -pic dostarcza narzędzi umożliwiających konstruowanie (logiczne składanie):

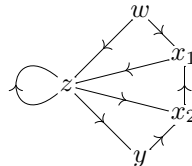
- diagramów macierzowych, w których do wybranego elementu grafu można odwołać się posługując się adresem kolumny/wiersza, na przecięciu których umieszczony jest dany element,
- diagramów mających postać drzewa,
- diagramów mających postać wielokątów regularnych,
- węzłów i połączeń,
- struktur dwukomórkowych.

Pokazane poniżej przykłady ilustrują wykorzystanie pakietu $\mathcal{X}\mathcal{Y}$ -pic do tworzenia diagramów:

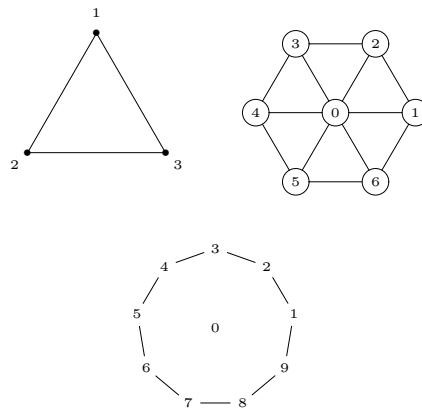
- ◊ diagram macierzowy [7]:



◇ diagram w postaci drzewa [8]:



◇ diagramy mające kształt wielokąta [8]:



Użycie pakietu $\mathcal{X}\mathcal{Y}$ -pic. W celu wykorzystania pakietu $\mathcal{X}\mathcal{Y}$ -pic do konstrukcji diagramów należy na początku dokumentu umieścić polecenia:

- formaty plain $\text{T}\mathcal{E}\text{X}$ i $\text{L}\mathcal{A}\text{T}\mathcal{E}\text{X}$:


```
\input{xy}
\xyoption{all}
```
- format $\text{L}\mathcal{A}\text{T}\mathcal{E}\text{X}2\text{e}$:


```
\usepackage[all]{xy}
```

Diagramy macierzowe

Na diagram można spojrzeć jako na obiekt, który ma strukturę macierzy. Takie diagramy nazywa się diagramami macierzowymi (*matrix like diagrams*). Rozdział ten poświęcony jest opisowi podstawowych konstrukcji znajdujących się w pakiecie $\mathcal{X}\mathcal{Y}$ -pic, pozwalających na budowanie takich diagramów. Diagramy macierzowe (jako struktura graficzna) mają własności:

- do poszczególnego elementu grafu można odwołać się przez adres kolumny/wiersza, na przecięciu których element ten jest umieszczony,
- elementy składające się na macierz diagramu są automatycznie wyrównywane w kolumnach i wierszach,
- każdy element może zostać połączony z innym za pomocą różnego rodzaju strzałek,
- strzałki mogą być łączone w pary, krzyżować się i otaczać inne elementy struktury diagramu,
- strzałki mogą być opisane za pomocą etykiet, które można dowiązać do określonego punktu na strzałce i przesunąć w jakimś kierunku.

Diagram typu macierzowego tworzony jest za pomocą polecenia `\xymatrix`, którego składnia ma postać:

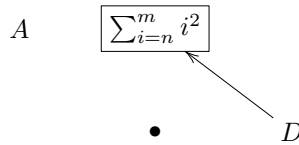
`\xymatrix <setup> { <species> }`

gdzie:

- `<setup>` oznacza zbiór przełączników (nazywanych @-wyrażeniami), które należy umieścić pomiędzy `\xymatrix` a nawiasem `{`; umożliwiają one zmianę szerokości kolumn i wysokości wiersza diagramu, rotację itp.,
- `<species>` oznacza elementy (obiekty), które zostaną umieszczone w kolumnach i wierszach:
 - elementy w wierszu są rozdzielane za pomocą znaku `&`,
 - poszczególne wiersze rozdzielane są znakiem `\\`.

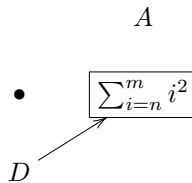
Przykłady (znaczenie poleceń, za pomocą których opisane są poszczególne diagramy, zostanie wyjaśnione w dalszej części artykułu):

(a)



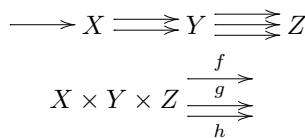
```
\xymatrix{
A & **+[F] {\sum_{i=n}^m {i^2}} \\
& & {\bullet} & D \ar[u]
}
```

(b)



```
\xymatrix @d
@R-\baselineskip @C-\baselineskip{
A & **+[F] {\sum_{i=n}^m {i^2}} \\
& & {\bullet} & D \ar[u]
}
```

(c)



```
$$
\xymatrix{\ar[r] & & & \\
X \ar@<2pt>[r] \ar@<-2pt>[r] & & Y & \\
& \ar@<4pt>[r] \ar[r] & & \\
& \ar@<-4pt>[r] & & Z}
$$
$$
\xymatrix{
X \times Y \times Z \\
Z \ar@<8pt>[r]^-{f} \\
& \ar@<-2pt>[r]^-{g} \\
& \ar@<-6pt>[r]_-{h} & & }
$$
```

Analiza powyższych przykładów diagramów macierzowych wskazuje, że:

- elementy diagramu składane są w taki sposób jak tekst matematyczny,
- elementy rysunku nie mogą rozpoczynać od nazwy makrodefinicji; makrodefinicje muszą zostać zamknięte w $\{\}$ (\bullet w przykładach (a) i (b)),
- wszystkie elementy (obiekty) w diagramie są centrowane; oddzielenie wierszy i kolumn w diagramie jest zawsze dostatecznie duże,
- elementy puste (na końcu lub na początku wiersza) mogą zostać opuszczone (zalecane jest jednak używanie $\{\}$ do wskazania takiego elementu),
- „ozdobniki Xy ” („ Xy-decorations ”) występujące przy elementach diagramu (np. $\ar[u1]$) pozwalają na rysowanie strzałek bez zmiany ogólnych ustawień,
- „modyfikatory Xy ” („ Xy-modifiers ”) umieszczone na początku elementu umożliwiają zmianę jego formatu i kształtu (np. $++[F]$ w przykładach (a) i (b)).

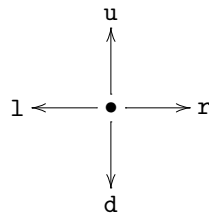
Strzałki. Logiczne powiązanie wybranych elementów diagramu w Xy-pic realizowane jest za pomocą **ścieżki** (*path*).

Do reprezentowania **ścieżek** w pakiecie Xy-pic służą **strzałki**. **Strzałka** jest ogólną nazwą obiektów graficznych łączących dwa wskazane elementy struktury macierzowej. W Xy-pic wszystkie strzałki muszą być określone razem z elementem, od którego nastąpi ich rysowanie (taki element nazywany jest **bazowym**). Każdemu poleceniu rysowania strzałki odpowiada jego własny element docelowy, znajdujący się w strukturze diagramu. Należy zauważyć, że element docelowy może być elementem bazowym i na odwrót.

W Xy-pic do rysowania strzałek służy polecenie \ar . Podstawowa wersja postać tej komendy ma następującą postać:

$$\ar[direction]$$

Zmienna *direction* opisuje kierunek strzałki, który określony jest przez kombinację znaków przestawionych na diagramie:



gdzie: *u* – do góry (*up*), *d* – na dół (*down*), *l* – w lewo (*left*) i *r* – w prawo (*right*). Jak widać znaki te reprezentują cztery podstawowe kierunki poruszania się. Np. wartość *rd* oznacza, że kierunek rysowania strzałki w prawo i na dół. Jeżeli w/w wartości pojawiają się *k*-razy to wskazuje to na *k*-krotne wydłużenie strzałki w kierunku wskazanym przez tę wartość. Można zatem powiedzieć, że zmienna *direction* w specyficzny sposób określa współrzędne końca strzałki.

Przykłady (poniżej pokazane są sekwencje poleceń, za pomocą których narysowano strzałki):

- ◇ $\ar[r]$ \longrightarrow
 $\xymatrix{ \{\} \ar[r] & \{\} }$
- ◇ $\ar[rr]$ \longrightarrow
 $\xymatrix{ \{\} \ar[rr] & \{\} & \{\} }$
- ◇ $\ar[rd]$
 \searrow
 $\xymatrix{ \{\} \ar[rd] & \{\} \\ & \{\} & \{\} }$
- ◇ $\ar[rdr]$ \searrow

```
\xymatrix{
{} \ar[rdr] & {} & {} \\
{} & {} & {} }
```

Jeżeli użyje się `\ar[direction]` bez wskazania obiektu docelowego, będącego elementem struktury diagramu, tzn. jeśli wyjdzie się poza obszar diagramu to zostanie zgłoszony błąd. Np. sekwencja poleceń:

```
\xymatrix{ {} \ar[rdr] & {} \\
{} & {} }
```

spowoduje błąd, ponieważ koniec strzałki, wskazany przez zmienną `rdr` znajduje się poza diagramem, który ma dwie kolumny i dwa wiersze (koniec strzałki pokazuje na element o współrzędnych (2,3)).

Style strzałek. Parametry, które charakteryzują strzałkę to: początek (*tail*), odcinek łączący (*shaft*) i grot (*head*). Określają one *styl* strzałki, tzn. sposób jej rysowania. Standardowa strzałka nie posiada początku, zakończona jest grotem, a linia łącząca jest linią ciągłą. Sposób jej rysowania można zmienić przez podanie co najmniej jednego z wymienionych powyżej parametrów. W celu określenia stylu strzałki należy przed zmienną `[direction]` w rozkazie rysowania strzałki `\ar` umieścić przełącznik `@style`. Wówczas polecenie rysowania strzałki przyjmie postać:

```
\ar@style[direction]
```

gdzie parametr `style` może mieć jedną z postaci:

- `variant{<tail><shaft><head>}`
- `variant{<head>}`
- `variant{<shaft>}`

Parametr `variant` przyjmuje wartości:

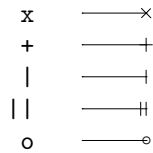
- `-` – pusty (nie jest to spacja) tzn. parametr `variant` nie występuje, co oznacza standardowy sposób kreślenia strzałki,
- `^` – początek i grot strzałki będą kreślone powyżej linii łączącej elementy,
- `-` – początek i grot strzałki będą kreślone poniżej linii łączącej elementy,
- `2` – podwojenie linii łączącej elementy,
- `3` – potrójnie linii łączącej elementy.

Wartości parametru `<shaft>` są następujące:

- `-` – linia pojedyncza ciągła ——— ,
- `=` – linia podwójna ciągła ===== ,
- `.` – linia pojedyncza kropkowana ,
- `:` – linia podwójna kropkowana ,
- `~` – linia pojedyncza falista ciągła ~~~~~ ,
- `~` – linia pojedyncza falista przerywana ~~~~~ ,
- `--` – linia pojedyncza przerywana - - - ,
- `-` – brak linii łączącej dwa elementy.

Poniżej pokazane są różne postaci początków (parametr `<tail>`) i grotów (parametr `<head>`) strzałek:

<code><</code>	———<	<code>></code>	———>
<code><<</code>	———<<	<code>>></code>	———>>
<code> <</code>	———<	<code>> </code>	———>
<code> <<</code>	———<<	<code>>> </code>	———>>
<code>(</code>	———(<code>)</code>	———)
<code>/</code>	———/	<code>//</code>	———#



Możliwe jest także tworzenie bardziej złożonych postaci parametru *style*:

- `variant{variant{<tail><shaft><head>}`
- `variant{<tail>variant{<shaft><head>}`
- `variant{<tail><shaft>variant{<head>}}`

Poniżej pokazane przykłady ilustrują użycie parametru *style*:



Warto zwrócić uwagę na przykłady pierwszy i ostatni. Pojawienie się w ostatnim przykładzie cyfry 2 przed opisem grotu strzałki tzn. przed `{>}` spowodowało zmianę jego kształtu.

Grot strzałki można rysować za pomocą różnych fontów. W celu zmiany używanego fonta należy przed poleceniem `\xymatrix` umieścić komendę postaci:

```
\SelectTips{font's family}{font's size}
```

gdzie:

- `font's family` - oznacza rodzinę fontów, które zostaną użyte do rysowania grotu strzałki,
- `font's size` - określa wielkość fonta.

Tabela przedstawia sposób rysowania grotu strzałki za pomocą różnych fontów:

Rodzina fontów	10	11	12
fonty xy	→	→	→
fonty computer modern	→	→	→
fonty Eulera	→	→	→

Działanie polecenia `\SelectTips` można ograniczyć do wybranego diagramu. Należy wówczas umieścić polecenia `\SelectTips` i `\xymatrix{ }` w nawiasach `{ }`. Ilustrują to poniższe przykłady:

◇ rodzina fontów cm:

```
A → C → B
{\SelectTips{cm}{11}
 \xymatrix{A \ar[r] & C
 \ar@{-3{>}}[r] & B}
}
```

◇ rodzina fontów eu:

```
A → C → B
```

```
{\SelectTips{eu}{11}
 \xymatrix{A \ar[r] & C \\
 \ar@{-3{>}}[r] & B}
}
```

Użycie komendy `\NoTips` powoduje powrót do ustawień domyślnych.

Strzałki kolorowe. Kolor rysowanej strzałki można zmienić za pomocą parametru:

```
@*{[color]}
```

gdzie zmienna `color` jest nazwą koloru użytego do kreślenia strzałki (np. `blue`, `red` itp.). Zastosowanie tej opcji wymaga użycia pakietu DVIPS.

Poniższy przykład pokazuje rysowanie strzałki w kolorze niebieskim:

```
◇ \xymatrix{A \ar@*{[blue]}[rrr] &&& B}
```

$A \xrightarrow{\text{niebieski}} B$

Zmiana grubości strzałki. Grubość linii użytej do kreślenia strzałki można zmienić za pomocą parametru:

```
@*{[|<dimen>]}
```

gdzie zmienna `dimen` określa powiększenie standardowej grubości linii. Zastosowanie tej opcji wymaga użycia pakietu DVIPS.

Poniższy przykład pokazuje rysowanie strzałki, której grubość jest powiększona o 5pt:

```
◇ \xymatrix{A \ar@*{[|<5pt>]}[rrr] &&& B}
```

$A \xrightarrow{\text{grubo}} B$

Etykiety strzałek. Strzałki można opisywać za pomocą etykiet. Przez etykietę rozumie się:

- pusty ciąg znaków,
- pojedyncze litery i cyfry,
- sekwencje liter i cyfr,
- proste formuły matematyczne (wówczas należy zamknąć je w nawiasach `{}`),
- dowolny element, obiekt.

Położenie etykiety względem strzałki określane jest za pomocą:

- `^` (daszek) – etykieta znajduje się nad strzałką, która jest skierowana w prawo, lub pod strzałką, gdy skierowana jest w lewo:

```
◇ \ar[r]^f \qquad \xrightarrow{f}
```

```
◇ \ar[l]^f \qquad \xleftarrow{f}
```

- `_` (podkreślenie) – etykieta zostanie umieszczona pod strzałką, jeżeli jest strzałka skierowana w prawo, lub nad strzałką, gdy strzałka skierowana jest w lewo:

```
- \ar[r]_f \qquad \xrightarrow{g}
```

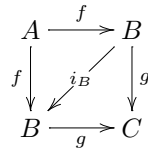
```
- \ar[l]_f \qquad \xleftarrow{f}
```

- `|` (pionowa kreska) – etykieta powoduje przerwanie („złamanie”) strzałki:

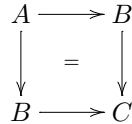
```
◇ \xymatrix{A \ar[r]|f & B}
```

$A \xrightarrow{f} B$

```
◇ \xymatrix{
 A \ar[d]_f \ar[r]^f & B \\
 \ar@{->}[dl] | {i_B} \ar[d]^g \\
 B \ar[r]_g & C }
```



Przerwanie strzałki pozwala na umieszczenie etykiety w dowolnym miejscu diagramu. Można to zrealizować przez użycie metody łamania niewidocznej strzałki, którą otrzymuje się za pomocą stylu @{}.



został narysowany za pomocą następującej sekwencji poleceń:

```

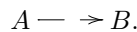
 $\begin{array}{ccc}
 A & \longrightarrow & B \\
 \downarrow & = & \downarrow \\
 B & \longrightarrow & C
 \end{array}$ 

```

Przerwanie strzałki bez umieszczania etykiety (*empty break*) można uzyskać za pomocą specjalnego polecenia \hole. Np. sekwencja

```
\xymatrix{A \ar[r]|\hole & B}
```

spowoduje wygenerowanie diagramu:



Ten sam efekt można uzyskać przerywając strzałkę za pomocą pustej etykiety (zawierającej tylko spację). Np. następujący ciąg poleceń:

```
\xymatrix{A \ar[r]|\mbox{\ } & B}
```

utworzy diagram: $A \text{---} \rightarrow B$.

W pakiecie Xy-pic etykieta umieszczana jest domyślnie w pozycji pionowej w połowie odległości pomiędzy środkami obiektów: bazowego i docelowego (dokładniej w tym miejscu znajduje się środek obszaru zawierającego etykietę), co pokazuje poniższy przykład:

```
A \ar[r]^{\theta} & B \otimes B
```

$$A \xrightarrow{\theta} B \otimes B$$

Domyślne położenie etykiety można zmienić za pomocą przesunięcia wstawianego pomiędzy ^, _ lub | a etykietą. Polecenie realizujące wspomnianą powyżej zmianę położenia jest postaci (jeszcze raz podkreślmy, iż wstawia się je przed etykietą):

```
place<translation>
```

gdzie:

- parametr place wskazuje na położenie etykiety względem rysowanej strzałki i przyjmuje wartości: _, ^ lub |,
- zmienna <translation> określa (opisuje) położenie środka obszaru, zawierającego etykietę i przyjmuje wartości:

- - Etykieta umieszczana jest na środku aktualnej strzałki:

```
A \ar[r]^{\theta} & B \otimes B
```

$$A \xrightarrow{\theta} B \otimes B$$

- <

Powoduje przesunięciem etykiety do punktu, w którym rozpoczyna się strzałka:

$A \ \backslash\text{ar}[r]_{<+} \ \& \ B$

$$A \xrightarrow{+} B$$

- >

Następuje przesunięcie etykiety do punktu, w którym kończy się strzałka:

$A \ \backslash\text{ar}[r]_{>+} \ \& \ B$

$$A \longrightarrow_{+} B$$

- << lub >>

Etykieta zostanie przesunięta do punktu, znajdującego się 3pt za początkiem lub przed końcem strzałki:

$A \ \backslash\text{ar}[r]_{>>+} \ \& \ B$

$$A \xrightarrow{+} B$$

Użycie większej liczby znaków <, > powoduje dalsze przesuwanie się etykiety w odpowiednim kierunku.

- (α)

Podanie czynnika skalującego w tej postaci powoduje przesunięcie etykiety do punktu, którego położenie na odcinku łączącym środek obszaru zawierającego element bazowy ze środkiem obszaru zawierającego element docelowy obliczane jest w ten sposób, że mnoży się długość wspomnianego odcinka przez czynnik skalujący (przy czym dla elementu bazowego $\alpha = 0$, dla elementu docelowego $\alpha = 1$, a $0 < \alpha < 1$ wskazuje na pozostałe punkty odcinka):

◊ $A \ \backslash\text{ar}[r]_{(0.3)+} \ \& \ B$

$$A \xrightarrow{+} B$$

◊ $A \ \backslash\text{ar}[r]_{(0.7)+} \ \& \ B$

$$A \xrightarrow{+} B$$

Czynnik skalujący można także umieścić za znakami: < lub >. W takim przypadku położenie etykiety obliczane jest w taki sposób jakby początkiem odcinka, którego długość brana jest w obliczaniu współrzędnej był element wskazany przez <, a końcem był element wskazany przez >. W poniższym przykładzie środek obszaru zawierającego etykietę zostanie umieszczony w punkcie, którego współrzędna zostanie obliczona w wyniku pomnożenia długości odcinka którego początkiem jest początek strzałki (na to wskazuje użycie <) a końcem środek obszaru zawierającego literę B przez wartość czynnika skalującego.

◊ $A \ \backslash\text{ar}[r]_{<(0.3)+} \ \& \ B$

$$A \xrightarrow{+} B$$

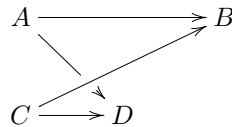
◊ $A \ \backslash\text{ar}[r]_{<(0.7)+} \ \& \ B$

$$A \xrightarrow{+} B$$

- Można także wskazać miejsca przecięcia się strzałki z wirtualnym odcinkiem, którego współrzędne końców określone są za pomocą $!\{t_1, t_2\}$, gdzie $\{t_1, t_2\}$ przyjmują takie wartości jak zmienna *direction* (tj. u, d, r i l itd.), a następnie umieszczenie we wskazanym miejscu jakiegoś elementu (np. pustego miejsca). Aby dobrze zrozumieć znaczenie tej konstrukcji należy uważnie przeanalizować podane poniżej przykłady jej użycia:

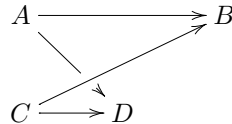
◊ w tym przykładzie strzałka o kierunku [dr] przecina się z wirtualnym odcinkiem, który opisany jest za pomocą $!\{[d], [rr]\}$ (jego końce mają współrzędne [d] (początek odcinka) i [rr] (koniec odcinka)), a w miejscu przecięcia umieszczony jest element pusty (\hole):

```
A \var[rr]
  \var[dr] !!\{[d];[rr]\hole
  & & B \\
C \var[r]
  \var[urr] & D
```



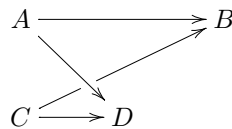
- ◊ w tym przykładzie zmieniony został jedynie sposób określenia wirtualnego odcinka, który opisany jest za pomocą `!{[rr],[d]}` (początek odcinka wskazany jest przez `[rr]` a koniec przez `[d]`), a wynik działania jest identyczny jak w przykładzie powyżej:

```
A \var[rr]
  \var[dr] |!{[rr];[d]}\hole
    & & B \\
C \var[r]
  \var[urr] & D
```



- ◊ w tym przykładzie z wirtualnym odcinkiem `!{[u],[r]}` przecina się strzałka o kierunku `[urr]` kreślona od elementu C, a otrzymany diagram jest identyczny z tymi z przykładów powyżej:

```
A \var[rr]
  \var[dr] & & B \\
C \var[r]
  \var[urr] |!{[u];[r]}\hole & D
```



Zakrzywienia. Strzałki łączące wybrane elementy diagramu mogą ulegać zakrzywieniu. Efekt ten uzyskuje się:

- za pomocą specjalnego stylu:

```
@/curving/
```

gdzie zmienna `curving` jest postaci:

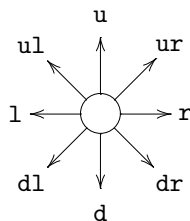
- `^<dimen>` - opisuje zakrzywienie „w górę”,
- `_<dimen>` - opisuje zakrzywienie „w dół”,

a parametr `<dimen>` określa wielkość zakrzywienia i może zostać opuszczony (wartość domyślna wynosi 5pt),

- przez podanie kierunku zakrzywienia w postaci:

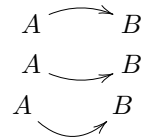
```
@(in,out)
```

gdzie parametr `in` definiuje kierunek wyjścia strzałki względem elementu bazowego, a `out` podaje kierunek dojścia strzałki do obiektu docelowego i obie te zmienne przyjmują one wartości pokazane na diagramie:

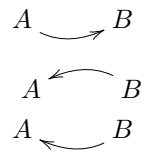


Pokazane poniżej przykłady ilustrują wymienione powyżej sposoby tworzenia zakrzywień strzałek:

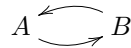
- ◇ `A \ar@{>}/[r] & B`
- ◇ `A \ar@{>}/_[r] & B`
- ◇ `A \ar@{>}/_1pc/[r] & B`



- ◇ `A \ar@{>}(dr,ru) & B`
- ◇ `A & B \ar@{>}(lu,ld)[l]`
- ◇ `A & B \ar@{>}(ld,lu)[l]`



- ◇ `A \ar@{>}(dr,ru)[r] & B \ar@{>}(lu,ld)[l]`



- ◇ `A \ar@{>}(ru,dr)[r] & B \ar@{>}(ld,lu)[l]`



- ◇ `A \ar@{>}(dr,dl)[rr] & & B`
`\ar@{>}(lu,ld)[l]+`



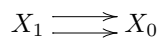
Strzałki równoległe. Rysowanie strzałek równoległych pomiędzy dwoma elementami diagramu polega na wykonaniu równoległego przesunięcia każdej strzałki względem położenia domyślnego. Służy do tego polecenie postaci:

```
\ar@{>}[translation][direction]
```

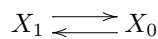
gdzie zmienna `translation` określa wielkość przesunięcia równoległego strzałki względem położenia domyślnego. Kierunek przesunięcia (góra, dół) zależy od znaku tej wielkości i kierunku strzałki.

Poniższe przykłady ilustrują wykorzystanie powyższej konstrukcji do rysowania strzałek równoległych:

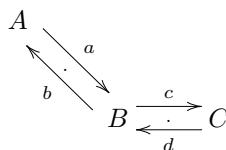
- ◇ `\xymatrix{X_1 \ar@{>+0.7ex}[r]`
`\ar@{>-0.7ex}[r] & X_0}`



- ◇ `\xymatrix{X_1 \ar@{>+0.7ex}[r] &`
`X_0 \ar@{>-0.7ex}[l]}`



- ◇ `\xymatrix{A \ar@{>1ex}(dr)^a_{.} \\\`
`& B \ar@{>1ex}(ul)^b \ar@{>1ex}(r)^c`
`& C \ar@{>1ex}(l)^d_{.} }`



◊ `\xymatrix{A \ar@/^/[r] \\ \ar@/^-1ex>[r] & B}`



Łączenie strzałek. W pakiecie Xy-pic możliwe jest łączenie ze sobą strzałek, które mają otoczyć wybrane elementy diagramu macierzowego. Wykonywane jest to za pomocą obrotów. W celu połączenia strzałek ze sobą należy pomiędzy `\ar` a `[<direction>]` należy wstawić:

`'<direction>[<target>]`

gdzie:

- `'` – apostrof pojedynczy odwrotny,
- parametr `<direction>` opisuje kierunek stycznej do okręgu po którym wykonywany jest obrót i przyjmuje wartości: `r`, `l`, `d`, `u` oraz ich kombinacje,
- zmienna `<target>` wskazuje na swego rodzaju miejsce przeznaczenia (zakończenia) dla obrotu wychodzącego z kierunku `<direction>`.

Obrót kończy się po wykonaniu 1/4 obrotu w stronę wskazaną przez `<target>`. Liczba sekwencji opisujący obroty tj. `'<direction>[<target>]` określona jest przez liczbę koniecznych do wykonania obrotów.

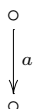
Zmienna `[target]` jest wymagana. Natomiast parametr `<direction>` nie jest obowiązkowy (można go pominąć). W zasadzie powinien on wystąpić jedynie dla pierwszego elementu ciągu sekwencji `'<direction>[<target>]` opisujących kolejne obroty (choć może pojawić się przy każdym). Jeżeli zmienna `<direction>` zostanie pominięta to przyjmuje się, że obrót wykonywany jest w prawo w kierunku zgodnym z ruchem wskazówek zegara. Kierunek wykonywanego obrotu względem poprzedniego można zmienić w następujący sposób:

- `^<direction>` – obrót wykonywany jest w kierunku przeciwnym do ruchu wskazówek zegara,
- `_<direction>` – obrót wykonywany jest w kierunku zgodnym z ruchem wskazówek zegara.

Domyślny promień obrotu wynosi 10pt i może zostać zmieniony przez wstawienie pomiędzy `'` a zmienną `<direction>` parametru `/R`, gdzie `R` jest długością promienia okręgu, po którym wykonywany jest obrót.

Poniższe przykłady pokazują sposoby użycia omawianej powyżej konstrukcji:

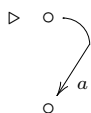
◊ w tym przykładzie diagram:



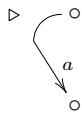
otrzymany za pomocą następującej sekwencji poleceń:

```
{\nonprefixing
\xymatrix{
{\circ} \ar [d] ^a \{\circ\}
}
```

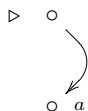
został poddany modyfikacjom, w których wykorzystana zostanie metoda łączenia strzałek (kierunki obrotów są wyznaczone kolejno przez `'r[d]`, `'l[d]` i `'rd[d]`):



```
{\nonprefixing
\xymatrix{
{\circ} \ar 'r[d] [d] ^a \{\circ\}
}
```

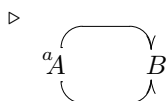


```
{\nonprefixing
\matrix{
{\circ} \ar{l[d][d] ^a \\\
{\circ}}
}
```



```
{\nonprefixing
\matrix{
{\circ} \ar{rd[d][d] ^a & \\\
{\circ}}
}
```

- ◊ w tym przykładzie wykonane są cztery obroty opisane przez 'u[r], ' [r] (górną strzałką), 'd[r] i ' [r] (dolną strzałką):



```
\matrix{
A \ar@<-2pt> ^a ' [r] [r]
\ar@<+2pt> 'd[r] ' [r] [r] & B}
}
```

Zmiana rozmiarów i obroty diagramu

Zmiana rozmiarów. W pakiecie Xy-pic umożliwia jest zmiana domyślnych rozmiarów kolumn, wierszy i całego diagramu macierzowego, a także jego elementów składowych. Można to zrobić za pomocą wstawianego pomiędzy \xymatrix i { przełącznika o postaci:

```
@spacing=<dimen>
```

gdzie:

- zmienna spacing przyjmuje wartości:

- ustalenie jednakowych rozmiarów dla kolumn i wierszy (za znakiem @ występuje znak =),
- R - ustalenie szerokości wiersza,
- C - ustalenie szerokości kolumny,
- M - ustalenie domyślnego marginesu dla obiektu,
- W - ustalenie domyślnej szerokości diagramu,
- H - ustalenie domyślnej wysokości diagramu,
- L - ustalenie marginesu etykiety,
- ! - wymuszenie ustawienia jednakowych rozmiarów dla kolumn i wierszy (alternatywna postać do @=),
- !O - zignorowanie wewnętrznego rozmiaru elementu,
- !R - wymuszenie ustawienia jednakowej wysokości wiersza (alternatywna postać do R),
- !C - wymuszenie ustawienia jednakowej szerokości kolumny (alternatywna postać do C),

- parametr <dimen> podaje wartości w/w rozmiarów.

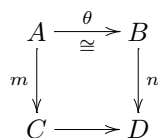
Uwaga: znak = może zostać zastąpiony przez:

- + - oznacza powiększenie aktualnego rozmiaru o <dimen> ,
- += - oznacza powiększenie aktualnego rozmiaru co najwyżej o <dimen> ,
- - - oznacza pomniejszenie aktualnego rozmiaru o <dimen> ,
- -= - oznacza pomniejszenie aktualnego rozmiaru co najmniej o <dimen> ,

Poniższe przykłady ilustrują zmiany rozmiarów diagramu:

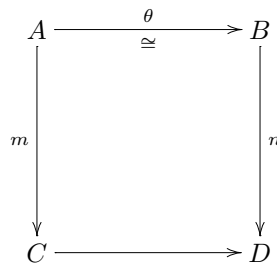
- ◊ tutaj pokazany jest diagram wyjściowy, które rozmiary zostały ustalone na podstawie parametrów domyślnych:

```
\xymatrix{A \ar[r]^{\theta} \cong \\ \ar[d]_m & B \ar[d]^n \\ C \ar[r] & D}
```



- ◊ w narysowanym poniżej diagramie rozmiary kolumn i wierszy są jednakowe i wynoszą 5pc (polecenie @!=5pc):

```
\xymatrix@!=5pc{A \\ \ar[r]^{\theta} \cong \\ \ar[d]_m & B \ar[d]^n \\ C \ar[r] & D}
```

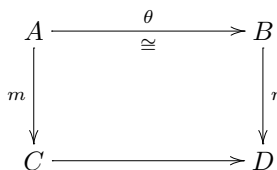


◊ w tym diagramie rozmiar kolumny wynosi 6pc (polecenie @C=6pc), a wysokość wiersza ustalona jest na 3pc (polecenie @R=3pc):

```

\xymatrix@R=3pc @C=6pc{A
  \ar[r]^{\theta}_\cong
  \ar[d]_m & B \ar[d]^n \\
C \ar[r] & & D}

```



Obroty diagramu. W pakiecie Xy-pic zostały zaprojektowane narzędzia pozwalające na obracanie diagramu. Obrót wykonywany jest wokół pierwszego elementu diagramu, tj. wokół obiektu znajdującego się w pierwszym wierszu i pierwszej kolumnie. Rotacja diagramu dokonywana jest za pomocą wstawianego pomiędzy \xymatrix i { polecenia:

```
@rotation
```

gdzie zmienna rotation określa kierunek i kąt obrotu i przyjmuje wartości:

- r - obrót o kąt 0,
- dr - obrót o kąt $-\pi/4$ (= rd),
- ur - obrót o kąt $\pi/4$ (= ru),
- l - obrót o kąt π
- lu - obrót o kąt $\frac{3}{4}\pi$ (= ul),
- ld - obrót o kąt $-\frac{3}{4}\pi$ (= ld),
- u - obrót o kąt $\pi/2$,
- d - obrót o kąt $-\pi/2$.

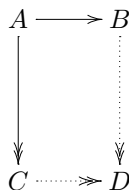
Poniższe przykłady pokazują obracanie pierwszego diagramu:

◊ diagram wyjściowy:

```

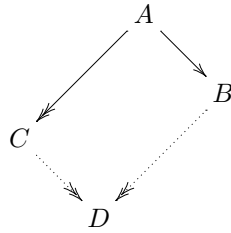
\xymatrix@R=4pc@C=2pc{
A \ar[r]\ar@{->>}[d] & B \\
  \ar@{.>>}[d] & \\
C \ar@{.>>}[r] & & D }

```



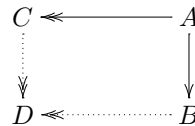
◊ diagram wyjściowy obrócony o kąt $-\pi/4$ (parametr @dr):

```
\xymatrix@dr@R=4pc@C=2pc{
A \ar[r]\ar@{->>}[d] & B \\
& \ar@{.>>}[d] & \\
C \ar@{.>>}[r] & & D }
```



◊ diagram wyjściowy obrócony o kąt $-\pi/2$ (parametr $@d$):

```
\xymatrix@d@R=4pc@C=2pc{
A \ar[r]\ar@{->>}[d] & B \\
& \ar@{.>>}[d] & \\
C \ar@{.>>}[r] & & D }
```



Zmiana kształtu i wielkości obszaru zawierającego element graficzny. Pakiet Xy-pic umożliwia zmianę wielkości i kształtu obszaru zarezerwowanego na umieszczony w nim element diagramu macierzowego. Do tego celu służy polecenie postaci:

```
*modifiers{text}
```

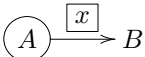


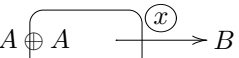

Zmienna `modifiers` opisuje w jaki sposób zmieni się format i kształt wspomnianego obszaru i przyjmuje wartości:

- ◊ `+` – powiększenie obszaru o $\langle 3pt \rangle$,
- ◊ `+<dimen>` – powiększenie obszaru o $\langle dimen \rangle$,
- ◊ `-` – pomniejszenie o $\langle 3pt \rangle$,
- ◊ `-<dimen>` – pomniejszenie o $\langle dimen \rangle$,
- ◊ `!` – element nie będzie centrowany,
- ◊ `[o]` – umieszczenie elementu w okręgu (musi wystąpić razem z `[F-]`, `[F=]`, `[F.]` lub `[F--]`),
- ◊ `[l]` `[r]` `[u]` `[d]` – przesunięcie w lewo, prawo, itp. o połowę szerokości obiektu lub obszaru (zależy to od położenia tej wartości w ciągu zmiennych `modifiers`),
- ◊ `[F-]` `[F=]` – umieszczenie w ramce pojedynczej lub podwójnej,
- ◊ `[F.]` `[F--]` – umieszczenie w ramce z kropek lub kresek,
- ◊ `[F-,]` `[F-:<dimen>]` – umieszczenie w ramce cieniowanej lub o zaokrąglonych wierzchołkach (promień zaokrąglenia jest równy $\langle dimen \rangle$).

Zmiennych `modifiers` może być więcej niż jedna. Wówczas umieszcza się je jedna za drugą. Parametr `text` zawiera opis obiektu, który umieszczony zostanie w modyfikowanym obszarze.

Pokazane przykłady ilustrują wykorzystanie omawianej powyżej konstrukcji:

```
\xymatrix{A
\ar[r]^*+[o][F-]{x} & B}
A \xrightarrow{\textcircled{x}} B
\xymatrix{**<30pt>\{A\}
\ar[r]^*+[o][F-]{x} & B}}
```


- ◇ `\xymatrix{**<10pt>[o][F-]{A}`
`\ar[r]^-**[F-]{x} & B}}`

- ◇ `\xymatrix{**<15pt>[F-:<5pt>]{A}`
`\oplus A`
`\ar[r]^**[o][F-]{x} & B}}`

- ◇ `\xymatrix{*[l]<15pt>[F-:<5pt>]{A}`
`\oplus A`
`\ar[r]^**[o][F-]{x} & B}}`

- ◇ `\xymatrix{*[l]<15pt>[F-:<5pt>][1]`
`{A \oplus A}`
`\ar[r]^**[o][F-]{x} & B}}`

- ◇ `\xymatrix{**<15pt>[F-:<5pt>][1]{A}`
`\oplus A`
`\ar[r]^**[o][F-]{x} & B}}`


Przesuwanie wierzchołków diagramu. Położenie wybranego wierzchołka diagramu można zmienić za pomocą przesunięcia równoległego. W tym celu należy umieścić przed opisem tego elementu polecenie:

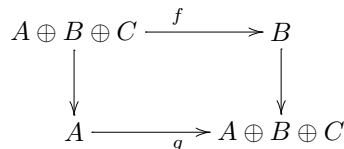
`**[direction]`

gdzie zmienna `direction` przyjmuje wartości:

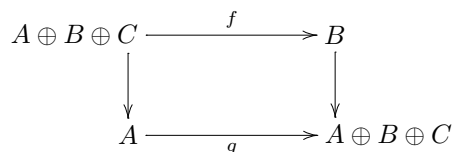
- `l` - powoduje przesunięcie w lewo o połowę szerokości obiektu,
- `r` - powoduje przesunięcie w prawo o połowę szerokości obiektu.

Pokazane przykłady ilustrują wykorzystanie omawianego parametru:

- ◇ `\xymatrix{`
`A \oplus B \oplus C \ar[r]^f`
`\ar[d] & B \ar[d] \\`
`A \ar[r]_g & A \oplus B`
`\oplus C}`



- ◇ `\xymatrix{`
`**[l]A \oplus B \oplus C \ar[r]^f`
`\ar[d] & B \ar[d] \\`
`A \ar[r]_g & **[r]A \oplus B`
`\oplus C}`

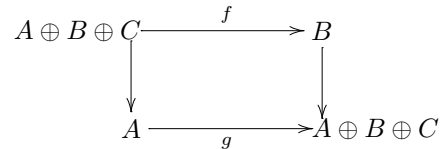


Należy tu zauważyć, że polecenia `**[direction]` i `*[direction]` dają różne efekty. Pokazuje to poniższy przykład.

```

\dymatrix{
  *[1]{A\oplus B \oplus C}
  \var[r]^f \var[d] & B \var[d] \\
  A \var[r]_g & *[r]{A\oplus B \\
  \oplus C}}

```



Aby zobaczyć różnice pomiędzy tymi diagramami należy zwrócić uwagę na elementy (1, 1) i (2, 2) obu diagramów.

Przykłady zastosowań pakietu $\mathcal{X}\mathcal{Y}$ -pic

W tym rozdziale pokazane zostały przykłady wykorzystania pakietu $\mathcal{X}\mathcal{Y}$ -pic do konstrukcji diagramów. Część z nich pochodzi z różnych publikacji, a kilka jest autorstwa autora tego artykułu. Każdy przykład składa się jakby z dwóch części: najpierw pokazany jest sam diagram, a pod nim sekwencje poleceń za pomocą których został wygenerowany. Uważna analiza diagramu pozwoli lepiej zrozumieć działanie pakietu $\mathcal{X}\mathcal{Y}$ -pic.

P1:

$$(d_0)!(d_0)!d_2^*d_1^* \xrightarrow{\tilde{d}_1=(d_0)!e^{d_1}d_1^*} (d_0)!d_1^*$$

```

\dymatrix{
  ( d_0)_!(d_0)_!d^*_2d^*_1
  \var[r]^{\tilde{d}_1}=(d_0)_!
  \epsilon^{d_1}d_1^*}
  & (d_0)_!d_1^*}

```

P2:

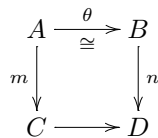
$$(d_0)!(d_0)!d_2^*d_1^* \xrightarrow{\tilde{d}_1=(d_0)!e^{d_1}d_1^*} (d_0)!d_1^*$$

```

\dymatrix{
  ( d_0)_!(d_0)_!d^*_2d^*_1
  \var[rrr]^{\tilde{d}_1}=(d_0)_!
  \epsilon^{d_1}d_1^*}
  & & (d_0)_!d_1^*}

```

P3:

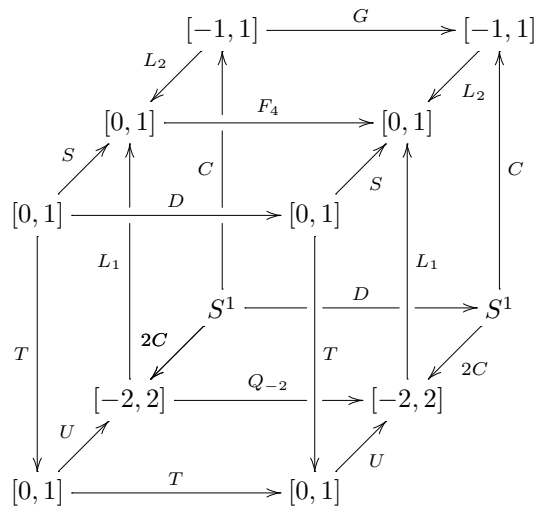


```

\dymatrix{
  A \var[r]^{\theta} \cong \var[d]_m & \\
  & B \var[d]^n \\
  C \var[r]} & D}

```

P4: Patrz [9]

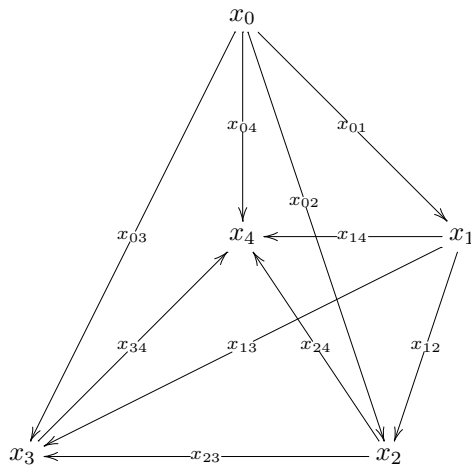


% ,,2 cube" matrix, adapted from
 % source by T. Scavo (18 July 1994)

```
\xymatrix@!=0.9pc{
%
& & [-1,1] \ar[rrr]^G \ar[dl]_{L_2} \\
& & & [-1,1] \ar[dl]^{L_2} \\
& & [0,1] \ar[rrr]^{F_4} \\
& & & & [0,1] \\
[0,1] \ar[rrr]^D \ar[ur]^S \\
& & & & \ar[ddd]_T \\
& & & [0,1] \ar[ur]_S \ar[ddd]^T \\
& & S^1 \ar[r]_{[r]'} \ar[r]_{[rr]} \ar[r]_{[rrr]} \\
& & & \ar@{}[rrr]^D \ar[dl]_{2C} \\
& & & \ar[dl]_{2C} \ar[u]'[uu] \ar[uu] \\
& & & \ar@{}[uuu]^C \\
& & S^1 \ar[uu]_C \ar[dl]^{2C} \\
& [-2,2] \ar[r]_{[rr]} \ar[r]_{[rrr]} \\
& & \ar@{}[rrr]^{Q_{-2}} \\
& & \ar[uu] \ar[uu] \ar@{}[uuu]^{L_1} \\
& & & [-2,2] \ar[uu]_{L_1} \\
& & & [0,1] \ar[rrr]^T \ar[ur]^U \\
& & [0,1] \ar[ur]_U \\
%
}

```

P5: Patrz [10]

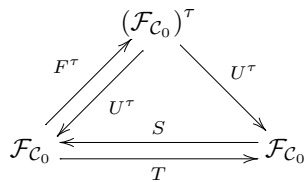


% "4 Simplex Model" adapted from
 % source by John Duskin, May 2001.

```

\matrix{
%
&&& x_0 \ar[ddd]|\{x_{04}\}
\ar[dddddl1l]|\{x_{03}\}
\ar[dddrrr]|\langle 0.4\rangle\{x_{02}\}
\ar[ddrrr]|\{x_{01}\} &&& \\\
&&&&&&& \\\
&&&&&&& \\\
&&& x_4 &&& x_1
\ar[ddd1]|\{x_{12}\}
\ar[ddd1l1l1l]|\{x_{13}\}
\ar[1l1l]|\{x_{14}\} &&& \\\
&&&&&&& \\\
&&&&&&& \\\
x_3 \ar[rrruuu]|\{x_{34}\}
&&&&& x_2
\ar[1l1l1l]|\{x_{23}\}
\ar[1luuu]|\{x_{24}\} &
%
}
    
```

P6: Patrz [9]



```

\matrix@!=2pc{
& \left(\mathcal{F}_{\mathcal{C}_0}\right)^{\tau}
\ar[dr]^{\mathcal{U}^{\tau}} \ar@<.7ex>[dl]^{\mathcal{U}^{\tau}}
& \mathcal{F}_{\mathcal{C}_0}
}
    
```

```

\var@<.7ex>[ur]^{F^\tau}
\var@<-.7ex>[rr]_T & &
{\mathcal F}_{\{\mathcal C\}_0}
\var@<-.7ex>[ll]_S }

```

P7: Patrz [9]

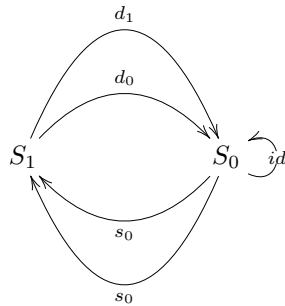
$$\begin{array}{ccc}
 \mathcal{F}_{S_0} & \xleftarrow{1} & \mathcal{D}_{\mathcal{F}} \xleftarrow{d_0^\#} \mathcal{F}_S \\
 & \xleftarrow{d_0^*} &
 \end{array}$$

```

{\nonprefixing
\xymatrix{
\mathcal{F}_{S_0} & & 
\mathcal{D}_{\mathcal{F}} & & 
\mathcal{F}_S & & 
\ar@/^{\text{1}} & & 
\ar@/_{d_0^\#} & & 
\ar@/^{\text{1}} & & 
\ar@/_{d_0^*} & & 
}

```

P8: Patrz [9]

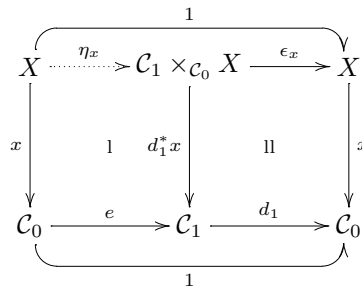


```

{\nonprefixing
\xymatrix@C=5pc{
S_1 \ar@/^4pc/[r]^{d_1}
\ar@/^2pc/[r]^{d_0} & 
S_0 \ar@(dr,ur)[]|{id}
\ar@/^2pc/[l]^{s_0}}
}

```

P9: Patrz [9]



```

\xymatrix@!=3pc{
X \ar@<-2pt> 'u[r] ' [rr]^{1} [rr]
\ar@{.}>[r]^{-\eta_x} \ar[d]_x
\ar@{}[dr]|{\rm 1} & & 
{\mathcal C}_1 \times_{\mathcal C_0} X
\mathcal{C}_0 \xrightarrow{e} \mathcal{C}_1 \xrightarrow{d_1} \mathcal{C}_0
}

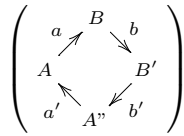
```

```

\ar[r]^-{\epsilon_x}\ar[d]_{d^*_1x}
\ar@{}[dr]|{\rm ll} & X \ar[d]^x \\
{\mathcal C}_0 \ar[r]^e
\ar@{>+2pt}> 'd[r] ' [rr]_1 [rr]
& {\mathcal C}_1 \ar[r]^{\{d_1\}} &
{\mathcal C}_0

```

P10: Patrz [9]

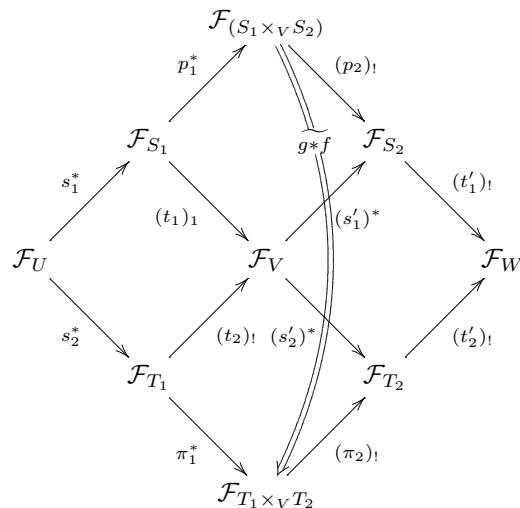


```

\left(
\def\objectstyle{\scriptstyle}
\def\labelstyle{\scriptstyle}
\vcenter{\xymatrix @-1.1pc @ur{
A \ar[r]^{\{a\}} & B \ar[d]^{\{b\}} \\
A'' \ar[u]^{\{a'\}} & B' \ar[l]^{\{b'\}}
}}
\right)

```

P11: Patrz [9]



```

{\nonprefixing
\xymatrix@!:=2pc{
& & \{\mathcal F\}_{\{(S_1 \times_V S_2)\}} \\
& & \ar[dr]^{\{p_2\}_!} \relax \\
\ar@/^2pc/@{>}[ddd]|(.25){
& & \widetilde{\{g * f\}} & & \\
& \{\mathcal F\}_{S_1} \ar[ur]^{\{p^*_1\}} \\
& \ar[dr]_{\{(t_1)_1\}} & & \\
& \{\mathcal F\}_{S_2} \\
& \ar[dr]^{\{(t'_1)_!\}} & & \\
& \{\mathcal F\}_U \ar[ur]^{\{s^*_1\}} \\
& \ar[dr]_{\{s^*_2\}} & & \\
& \{\mathcal F\}_V \ar[ur]_{\{(s'_1)^*\}} \\
& \ar[dr]_{\{(s'_2)^*\}} & &
}

```

```

{\mathcal F}_W \qquad \qquad \qquad \backslash\backslash
& \{\mathcal F\}_{T_1} \ \text{ar}[\text{dr}]_{\{\pi^*_1\}}
\ \text{ar}[\text{ur}]_{\{(t_2)_!\}} & &
{\mathcal F}_{T_2}
\ \text{ar}[\text{ur}]_{\{(t'_2)_!\}} & \qquad \qquad \backslash\backslash
& & \{\mathcal F\}_{T_1 \times V T_2}
\ \text{ar}[\text{ur}]_{\{(\pi_2)_!\}} & & \backslash\backslash
}
}

```

P12: Przykładowe diagramy, występujące w równaniach metody VU-CCSD/R (*Valence Universal - Coupled Cluster Single Double*) i odpowiadające im wyrażenia algebraiczne.

◇

$$\begin{array}{c} \rho \\ | \\ \text{---} \otimes \\ | \\ \alpha \end{array} = (2l_\alpha + 1)^{-1} f_\rho^\alpha$$

```

$\begin{array}{c}
{\nonprefixing}
\matrix@R=3ex{*0{}}
\ar@{-}[d]_{\rho} & & \backslash\backslash
{*0{}}\ar@{-}[d]_{\alpha}
\ar@{--}[r] & \{\otimes\} & \backslash\backslash
{*0{}}
\end{array}
$

```

$$\begin{array}{c} \rho \\ | \\ \text{---} \otimes \\ | \\ \tau \\ \text{====} \\ | \\ \alpha \end{array} = \sum_{\tau} (2l_\tau + 1)^{-1} f_\rho^\tau R_\tau^\alpha$$

```

$\begin{array}{c}
{\nonprefixing}
\matrix@R=3ex{
{*0{}}\ar@{-}[d]_{\rho} & & \backslash\backslash
{*0{}}\ar@{-}[d]_{\tau}
\ar@{--}[r] & \{\otimes\} & \backslash\backslash
{*0{}}\ar@{-}[d]_{\alpha}
\ar@{=} [r] & & \backslash\backslash
{*0{}} \quad \backslash\backslash
}
\end{array}
$

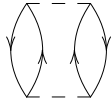
```

◇

```
f_{\rho}^{\tau} R_{\tau}^{\alpha}
\end{array}$
```

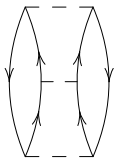
P13: Przykłady diagramów występujących w rachunku zaburzeń Møller'a-Plesset'a:

◊ diagram drugiego rzędu:



```
$\begin{array}{c}
{\nonprefixing}
\xymatrix@R=8ex{
  *0{\ar@{-}@/_/[d]!U| -{
    \object@{>}}\ar@{-}@^/[d]|
    -{\object@{<}} \ar@{--}[r]
  }
  & *0{\ar@{-}@^/[d]| -{
    \object@{>}} \ar@{-}@/_/[d]|
    -{\object@{<}}\ \
  }
  *0{\ar@{--}[r] & *0{ }
}
\end{array}$
```

◊ diagram trzeciego rzędu:



```
$
{\nonprefixing}
\xymatrix@R=4.35ex@C=3.75ex{
  *0{\ar@{-}@/_/[dd]| -{
    \object@{>}}
    \ar@{-}@^/[dd]|(0.3){
    \object@{<}}|(0.7){
    \object@{<}}
    \ar@{--}[r]
  }
  & *0{\ar@{-}@^/[dd]| -{
    \object@{>}}
    \ar@{-}@/_/[dd]|(0.3){
    \object@{<}}|(0.7){
    \object@{<}} & \ \
    *++{\ar@{--}[r]*++{ } & & \ \
  }
  *0{ \ar@{--}[r] & *0{ } & & }
}
$
```

Podziękowania

Autor pragnie wyrazić podziękowanie Joli Szematyńskiej za wnikliwe przeczytanie tekstu artykułu i cenne uwagi.

Literatura

- [1] Paul Burchard (burchard@pobox.com), *User's Guide to the Diagram Environment, Version 5*, 1998
<http://www.ctan.org/tex-archive/macros/latex2e/contrib/supported/pb-diagram/>,
- [2] Thorsten Ohl, feynMF *Drawing Feynman Diagrams with L^AT_EX and Metafont*, 1996
<http://www.ctan.org/tex-archive/macros/latex/contrib/supported/feynmf/>,
- [3] Paul Taylor (pt@dcs.qmw.ac.uk), *Commutative Diagrams in T_EX (version 4)*, 1994
<http://www.dcs.qmul.ac.uk/~pt/diagrams/>,
- [4] Pedro Quaresma de Almeida (pedro@mat.uc.pt), *Logical Specification of Commutative Diagrams in a L^A(T_EX) Document*, 2002 <http://www.ctan.org/tex-archive/macros/generic/diagrams/dcpic/>
- [5] Michael Barr (barr@barrs.org), *A new diagram package*, 2002 <ftp://ftp.math.mcgill.ca/pub/barr/diagxy.zip>
- [6] *Xy-pic Home Page*: <http://www.tug.org/applications/Xy-pic/Xy-pic.html>,
<http://www.brics.dk/~krisrose/Xy-pic.html>, <http://www.maths.mq.edu.au/~ross/Xy-pic.html>,
- [7] Kristoffer H. Rose (krisrose@brics.dk), *Xy-pic User's Guide* Version 3.7, 1999,
- [8] Kristoffer H. Rose and Ross Moore (ross@mpce.mq.edu.au), *Xy-pic Reference Manual* Version 3.7, 1999,
- [9] Mohammed Alsani (alsani@math.buffalo.edu), *Examples on Typesetting Commutative Diagrams Using Xy-pic*, 2001,
- [10] John W. Duskin *Simplicial Matrices and the Nerves of Weak n-Categories I: Nerves of Bicategories*, *Theory and Applications of Categories*, Vol. **9**, No. **10**, (2002), 198–308.
- [11] Romuald Słupski (rbs@uni.torun.pl) *Zastosowanie pakietu Xy-pic do konstrukcji diagramów w teorii wielociałowego rachunku zaburzeń (MBPT) i teorii sprzężonych klasterów (CC) – seminarium Zakładu Teorii Układów Wieloelektronowych, Wydział Fizyki, Astronomii i Informatyki Stosowanej UMK*, 2002.