

Poszukiwanie węzłów idealnych

Piotr PIERAŃSKI

Instytut Fizyki Molekularnej, Polska Akademia Nauk
ul. Smoluchowskiego 17/19, 60-179 Poznań

ⁱ
Politechnika Poznańska, Instytut Fizyki
ul. Piotrowo, 60-965 Poznań

Otrzymano 20 grudnia 1996 roku

Streszczenie. W artykule przedstawiono opracowany przez autora algorytm poszukiwania optymalnych kształtów węzłów. W swym kształcie optymalnym węzeł wymaga do wykonania najkrótszego odcinka liny. Ponadto zaprezentowano kształty optymalne kilku węzłów pierwszych.

Słowa kluczowe: teoria węzłów, węzły, węzły idealne

1. Wprowadzenie

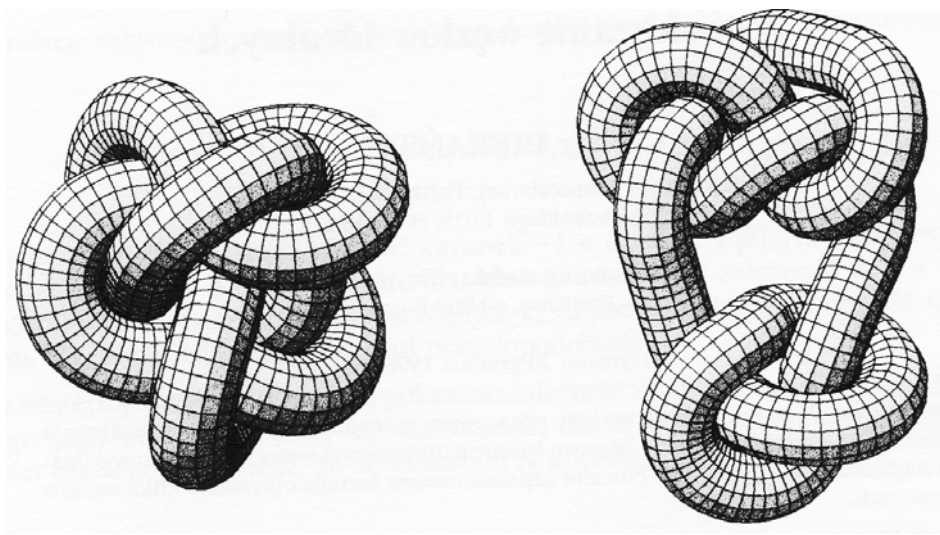
Weźmy kawałek liny, popłączmy ją w dowolny sposób i połączmy końce. Otrzymaliśmy obiekt, którego matematyczna idealizacja – *nieprzecinająca się sama z sobą, zamknięta krzywa w przestrzeni K* – nazywana jest *węzłem*. Z punktu widzenia teorii węzłów każda inna forma utworzonego już węzła uzyskana przez transformacje nie zrywające liny (zgniatanie, rozciąganie itp.) to ten sam węzeł [1].

Węzeł, który można przeformować tak, że będzie składał się z dwu części połączonych jedynie dwoma odcinkami liny, nazywany jest *węzłem złożonym* (zob. rys. 1).

Węzły nie dające się rozłożyć w ten sposób, nazywane są węzłami *pierwszymi* lub *prostymi*. Węzły pierwsze klasyfikujemy według minimalnej liczby skrzyżowań liny widzianych w ich rzutach na płaszczyznę (poszukując minimalnej liczby skrzyżowań możemy oczywiście nie tylko zmieniać płaszczyznę rzutowania, ale też dokonywać każdej, nie zmieniającej jego typu deformacji węzła).

Liczba różnych węzłów odnajdywanych w klasach węzłów pierwszych o k skrzyżowaniach rośnie, a poza przedziałem $k < 5$, gwałtownie z k . Jak łatwo sprawdzić, istnieje tylko jeden węzeł bez żadnego skrzyżowania. W swej najprostszej formie ma on kształt okręgu. Klasy dla $k = 1$ i $k = 2$ są puste – nie można utworzyć węzła wyłącznie z jednym lub dwoma skrzyżowaniami liny. Przy $k = 3$ i $k = 4$ mamy po jednym węzle. Dla $k = 5, 6, 7, 8, 9, 10, 11$ i 12 znajdujemy ich odpowiednio 2, 3, 7, 21, 49, 165, 552, 2176 (zauważmy, że tabele węzłów pierwszych zawierają tylko jeden z pary węzłów będących swymi lustrzanymi odbiciami, nawet jeśli nie są to węzły równoważne). Czytelnika zainteresowanego historią teorii węzłów dosyłamy do *Dodatku A* w monografii J. Przytyckiego [2].

W dalszej części zajmować się będziemy wyłącznie węzłami pierwszymi, nazywając je w skrócie węzłami.



Rys. 1. Węzeł złożony daje się rozłożyć na dwa węzły połączone tylko dwoma odcinkami liny. W przypadku przedstawionym na rysunku, węzły składowe są węzłami pierwszymi: 3_1 i 4_1 .

2. Problem optymalnego kształtu węzła

Przyjmijmy, że lina użyta do wiązania węzłów ma przekrój kołowy o średnicy D i będąc nieskończenie twarda jest jednocześnie całkowicie elastyczna, tzn. daje się bez trudu zgiąć nie zmieniając swej średnicy. Zadajmy nietrywialne pytanie:

Jaka jest minimalna długość L liny o średnicy D , przy której możliwe jest jeszcze zawiązanie wybranego węzła?

Pytając o minimalną długość liny pytamy w gruncie rzeczy o optymalny, z punktu widzenia zużycia liny, kształt danego węzła. Problem ten został niedawno postawiony przez grupę naukowców z Uniwersytetu w Lozannie badających węzły tworzone przez cząsteczki DNA [3]. Autorzy cytowanej pracy proponują, by węzły w swych optymalnych kształtach nazywane były węzłami *idealnymi* i przedstawiają kilka węzłów idealnych, podając też stosunek długości L liny użytej do ich zawiązania do jej średnicy D .

Poszukujący węzłów idealnych algorytm Monte Carlo użyty przez autorów pracy [3] jest złożony i czasochłonny. Celem prezentowanej tu pracy jest zaproponowanie innego, prostszego i – jak się wydaje – znacznie szybszego algorytmu. Algorytm ten stanowi numeryczną symulację procesu fizycznego, który można by określić jako *nadmuchiwanie węzła*.

3. Algorytm nadmuchiwania węzła

Wyobraźmy sobie, że węzeł wykonany jest nie z liny, lecz idealnie śliskiej rurki o średnicy, która może być w ciągły sposób zmieniana w procesie, który z kolei możemy określić jako nadmuchiwanie. Zakładamy przy tym, że średnica rurki jest zawsze jednorodna wzdłuż całego węzła (w realnym procesie nadmuchiwania, np. dętki rowerowej, warunek ten nie jest oczywiście spełniony). Nadmuchiwanie rurki, z której wykonano węzeł, prowadzi do zmian jego konfiguracji – węzeł sam poszukuje konfiguracji optymalnej, tzn. takiej, w której objętość zawarta wewnątrz rurki osiągnie swe maksimum.

Algorytm składa się z kilku części. Poniżej opiszemy je w kolejności, w jakiej zostają użyte, przedstawiając ogólne zasady ich działania, a pomijając mniej istotne szczegóły realizacyjne. Program realizujący opisany algorytm został napisany w języku Turbo Pascal 7.0 [4]. Rysunki wykonano przy użyciu procedur graficznych pakietu Maple V.4.

3.1. Definiowanie węzła

W opracowanym programie wiązanie węzła wykonywane jest w środowisku graficznym przy użyciu myszy i sprowadza się do przerysowania w formie łamanej rysunku wybranego węzła wziętego np. z tablicy opracowanej przez Dale Rolfsena (zob. dodatek *Tablice Węzłów* w [2]). Płaszczyzna ekranu monitora to płaszczyzna (x, y) . Użycie lewego i prawego klawisza myszy determinuje, czy dany odcinek węzła widziany na ekranie w rzucie na płaszczyznę (x, y) , przechodzi nad czy też pod płaszczyzną (x, y) , co pozwala odtworzyć przestrzenne ułożenie krzyżujących się odcinków węzła. Podczas rysowania tworzone są tablice współrzędnych (x, y, z) punktów ułożonych wzdłuż rysowanej łamanej. Odpowiednia procedura kończąca rysowanie węzła przeskalowuje współrzędne wygenerowanych punktów tak, by całkowita długość łamanej po wyjściu z procedury rysowania węzła wynosiła $L = 1$. Wielkość N zależy od typu i sposobu rysowania wybranego węzła i jest rzędu 100. Dla węzłów bardziej złożonych N jest większe niż dla węzłów, które można naszkicować łamaną o zaledwie kilku odcinkach.

3.2. Procedury korygujące odległości między cząstkami

W swej numerycznej reprezentacji węzeł składa się z N cząstek powiązanych w zamknięty łańcuch o całkowitej długości 1. Odległość między cząstkami będącymi najbliższymi sąsiadami w łańcuchu, tj. tymi, których indeksy różnią się o 1, nie może przekraczać wartości $l = 1/N$, choć w procesie poszukiwania kształtu idealnego może być chwilowo od niej mniejsza. Możemy więc sobie wyobrazić, że sąsiadujące w łańcuchu cząstki powiązane są N nierozciągliwymi nitkami, zwanymi dalej wiązaniami, każda o długości l . W procesie poszukiwania optymalnego kształtu węzła odległości te są kontrolowane i korygowane. Gdy odległość między dwiema cząstkami przekracza długość wiązania, ich położenia zostają skorygowane tak, by po korekcie odległość ta wynosiła dokładnie l . Procedura kontrolująca odległości między cząstkami sąsiadującymi w łańcuchu, nazwijmy ją SL, sprawdza w jednej pętli długości wszystkich N wiązań. W celu uniknięcia efektów ubocznych, sprawdzanie rozpoczyna się od losowo wybranej cząstki i losowo przebiega w prawo (w stronę rosnących indeksów) lub w lewo. Jeśli w procedurze SL napotkane zostanie wiązanie zbyt

długie, zostaje ono odpowiednio skrócone, przy czym skorygowaniu ulegają położenia obu cząstek – symetrycznie, względem ich środka ciężkości, wzdłuż linii łączącej ich środki. Procedura taka, jak łatwo zauważyć, narusza poprawność długości wiązań sąsiednich, co powoduje, że po wykonaniu pełnej pętli, długość łańcucha nie zostaje skorygowana dokładnie do wymaganej wartości 1. Okazuje się jednak, że ze względu na łamaną formę łańcucha jej jednokrotne wykonanie (przy nieobecności innych procedur korygujących) skraca nieco łańcuch powodując, że w kolejnych iteracjach jego długość zbiega się asymptotycznie do pewnej wartości nie większej od 1 (wartość ta może być mniejsza od jedności, bowiem, jeśli w wyjściowym łańcuchu są luźne wiązania, tzn. krótsze od $1/N$, to procedura SL pozostawi je w stanie nienaruszonym, chyba, że któraś z powiązanych takim wiązaniem cząstek wchodzi w skład innego, zbyt długiego wiązania).

Drugą z wartości, która musi być kontrolowana w procesie nadmuchiwania węzła, jest odległość d_{ij} między cząstkami i oraz j , które nie są najbliższymi sąsiadami w łańcuchu, ale które mogą zbliżyć się do siebie wskutek jego ułożenia w przestrzeni. Odległość ta nie może być mniejsza niż D – przyjęta średnica liny, z której wykonano węzeł. Jeśli wybierzemy cząstkę i -tą, to sprawdzenia wymagają odległości do wszystkich cząstek, których indeks j jest o $\mathbf{Round}(D/l)$ większy lub mniejszy od i (cząstki o indeksach j bliższych i wprawdzie znajdują się w odległościach w mniejszych niż D , ale nie oznacza to, że dwa odcinki węzła uległy przekryciu).

Z fizycznego punktu widzenia działanie procedury sprawdzającej odległości między cząsteczkami wchodzącymi w skład różnych odcinków węzła (nazwijmy tę procedurę RO) sprowadza się do pilnowania, by nigdzie w węźle nie pojawiało się przekrycie sąsiadujących z sobą w przestrzeni segmentów liny. Opracowana bez wyobraźni procedura RO może być bardzo czasochłonna. Czas jej wykonywania może rosnać z kwadratem N .

Wydatne skrócenie czasu obliczeń uzyskuje się przez usprawnienie procesu kontrolowania pojawiających się przekryć. W opracowanym przez autora programie zdefiniowano dodatkową tablicę dwuwskaźnikową (nazwijmy ją NN), w której i -tym wierszu przechowywane są indeksy cząstek znajdujących się w odległości mniejszej niż $(1+\epsilon)D$. Wartość parametru ϵ jest ustalana doświadczalnie. Tablica zawiera N wierszy. Jak łatwo sprawdzić, długość jej wierszy nie musi być większa niż $\mathbf{Round}(\pi D/l)$ – jest to liczba cząstek w segmencie węzła oplatającego kółko innego jego segment. Tablica NN jest co pewien, eksperymentalnie dobrany czas, zerowana i aktualizowana procedurą FN.

Sprawdzanie odległości między cząstkami rozpoczyna się od losowo wybranej cząstki i biegnie, podobnie jak to się czyni w procedurze sprawdzania długości wiązań, w prawo lub w lewo wzdłuż łańcucha. Wykorzystanie opisanej wyżej tablicy skraca wydatnie czas obliczeń, bowiem dla i -tej cząstki wystarczy sprawdzić tylko te cząstki, których indeksy są zapisane w i -tym wierszu – przerywając sprawdzanie po natrafieniu na pierwsze 0. Oczywiście, aktualizacja FN tablicy NN trwa znacznie dłużej, bo tu, dla uniknięcia niespodzianek, sprawdzana jest każda para cząstek (o indeksach różniących się o więcej niż $\mathbf{Round}(D/l)$).

Jeśli w procedurze sprawdzania odległości między cząstkami okazuje się, że para cząstek (i, j) znajduje się bliżej niż D , cząstki te są symetrycznie rozsuwane, po linii łączącej ich środki, na odległość D . Zauważmy, że procedura ta powoduje z reguły, iż wiązania $(i-1, i)$, $(i, i+1)$, $(j-1, j)$ oraz $(j, j+1)$ ulegają wydłużeniu, co może spowodować

przekroczenie wartości $1/N$. Błędy tego rodzaju zostają skorygowane podczas kolejnych kontroli długości wiązań.

3.3. Procedura przesuwu cząstek wewnątrz węzła

W procesie testowania programu stwierdzono, że można wydatnie przyspieszyć jego działanie wprowadzając dodatkową, cyklicznie wykonywaną procedurę SK, podczas której wszystkie cząstki zostają przesunięte wzdłuż łączących je wiązań o odległość niewspółmierną z l , np. l/π . Przesunięcie takie, widoczne jako płynięcie łańcucha cząstek wewnątrz niemal nieruchomej przestrzennej struktury węzła, utrudnia powstanie zaklinowań wynikających z jego dyskretnej natury. Jak łatwo sprawdzić, procedura ta powoduje też zacieśnianie się węzła, ponieważ podczas kolejnych przesunięć ścinane są jego zakręty. Zauważmy, że w procesie poszukiwania kształtu optymalnego konieczna bywa zmiana kierunku przesunięcia lub jego zatrzymanie.

3.4. Poszukiwanie kształtu optymalnego

Opisane wyżej procedury SL, RO FN i SK stanowią jądro programu, w którym symulowany jest proces nadmuchiwania węzła do maksymalnej objętości. W procesie tym węzeł zmienia swoją konfigurację nie zmieniając typu.

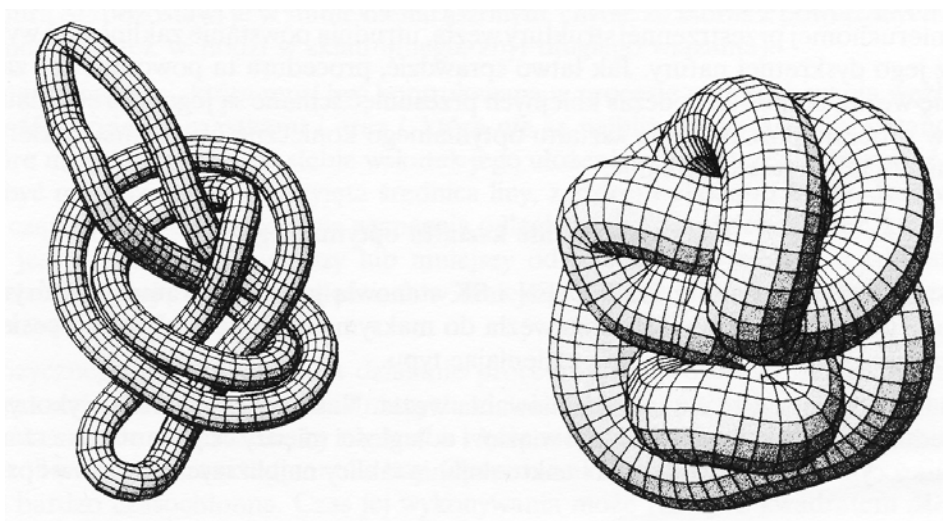
Symulacja rozpoczyna się od zdefiniowania węzła. Następnie cyklicznie wykonywane są procedury SL kontrolujące długości wiązań i odległości między cząstkami oraz rzadziej, ale również cyklicznie, procedury FN uaktualniania tablicy najbliższych sąsiadów i przesunięcia SK.

Jeśli podczas cyklicznie wykonywanych procedur RO i SL okazuje się, że sumaryczna długość łańcucha cząstek spadła poniżej $1+\delta$, gdzie δ oznacza dobierany doświadczalnie parametr, zwiększana jest wartość D . Zwiększenie średnicy D może prowadzić do pojawienia się przekręć wykrywanych i usuwanych następnie przez procedurę RO. Przekroczenia długości wiązań są usuwane przez procedurę SL itd.

Proces powiększania średnicy cząstek jest początkowo bardzo szybki, w miarę jednak, jak węzeł staje się coraz ciaśniejszy, ulega on spowolnieniu. Zmiany konfiguracji węzła monitorowane są graficznie. Kolejne fazy zmian jego konfiguracji są zbierane na dysku i na żądanie mogą zostać odtworzone w formie krótkiej animacji pozwalającej się zorientować, czy zmiany konfiguracji rzeczywiście ustały czy też tylko uległy spowolnieniu. Zauważmy, że zmiany konfiguracji są tym szybsze, im mniejsza jest liczba cząstek, z których zbudowany jest węzeł. Korzystne jest więc rozpoczęcie procesu poszukiwania kształtu optymalnego z małą (ale nie zbyt małą) liczbą cząstek i jej zwiększenie w sytuacji, gdy ustana już zmiany konfiguracji. Nagłe zwiększenie liczby cząstek, np. ich podwojenie, wymaga skokowego zmniejszenia uzyskanej już maksymalnej średnicy D tak, by w węźle wytworzył się pewien luz pozwalający na skorygowanie jego formy.

Opisany wyżej algorytm nie jest na tyle doskonały, by mógł samodzielnie znaleźć optymalny kształt każdego węzła. Jego użycie wymaga kontroli i eksperymentalnego doboru parametrów symulacji. Z pewnością też może zostać jeszcze znacznie ulepszony. Mimo tych niedoskonałości prezentujemy go jednak, bowiem uzyskane wyniki wydają się lepsze od opisanych w [3]. W przypadku paru węzłów udało się nam odnaleźć ich inne,

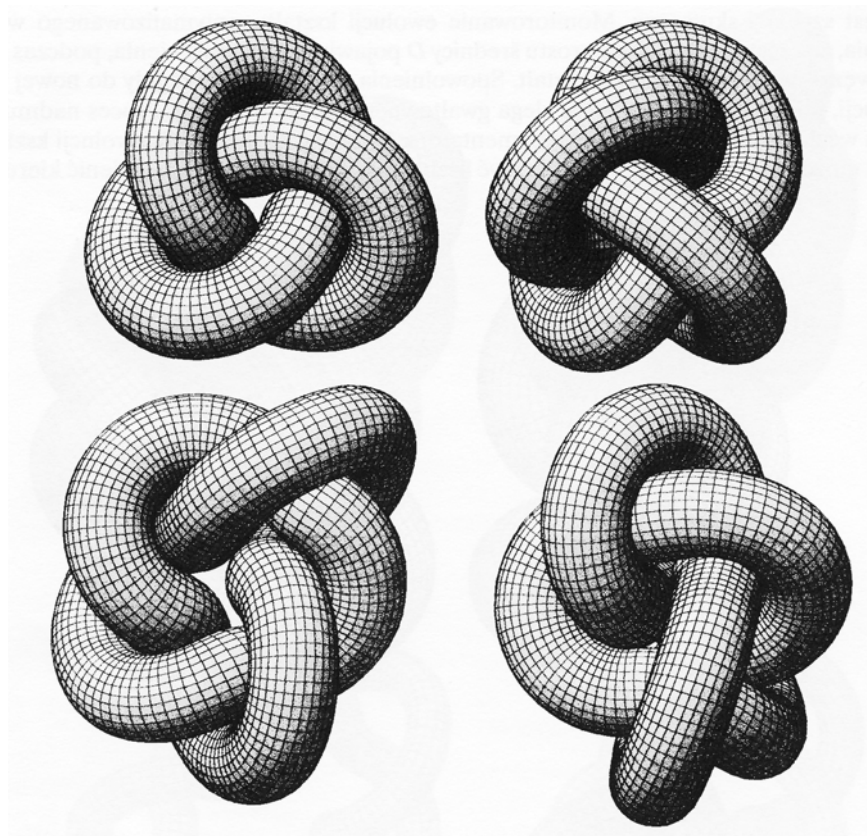
wygodniejsze formy. W poszukiwaniu optymalnego kształtu węzła opracowany algorytm powinien być w stanie wykonać czynność zwaną w życiu codziennym rozplątywaniem. Rys. 2 przedstawia wynik przeprowadzonego testu, w którym konfiguracja wyjściowa miała formę zaplątanego węzła trywialnego. Jak widać, algorytm nadmuchiwania rozsupłał tę konfigurację bez trudu.



Rys. 2. Algorytm nadmuchiwania węzła jest w stanie rozsupłać węzeł trywialny. Po lewej – konfiguracja wyjściowa; po prawej – stadium, w którym zawiązane części zostały wysunięte.

4. Optymalne kształty kilku węzłów

Zanim przedstawimy wyniki otrzymane przy użyciu opisanego wyżej algorytmu, zastanówmy się, czy dla jakiegokolwiek węzła znany jest ściśle jego kształt optymalny. Przejrzenie tablic węzłów nie daje powodów do optymizmu. Zauważmy jednak, że w tablicach tych nie ma węzła trywialnego złożonego z jednego kawałka liny, której końce zostały połączone w pętlę bez jakichkolwiek dodatkowych manipulacji. Jest oczywiste, że optymalnym kształtem dla węzła trywialnego jest okrąg. Jeśli założymy, że ma on długość 1, to maksymalny promień liny, z jakiej można go utworzyć, wynosi $r = 1/2\pi$. Poszukiwana wartość stosunku L/D wynosi więc w tym przypadku π . Jest to wartość minimalna – dla każdego innego węzła stosunek ten musi być większy. Odnotujmy, że podana bez komentarza przez autorów pracy [3] wartość 6,3, otrzymana dla splotu dwu węzłów trywialnych, wynosi, jak łatwo się przekonać, dokładnie 2π .



Rys. 3. Optymalne kształty węzłów 3_1 i 4_1 (góra) oraz 5_1 i 5_2 (dół). Odpowiadające im wartości ilorazu L/D wynoszą odpowiednio: 16,4 i 21,0 oraz 24,0 i 24,7.

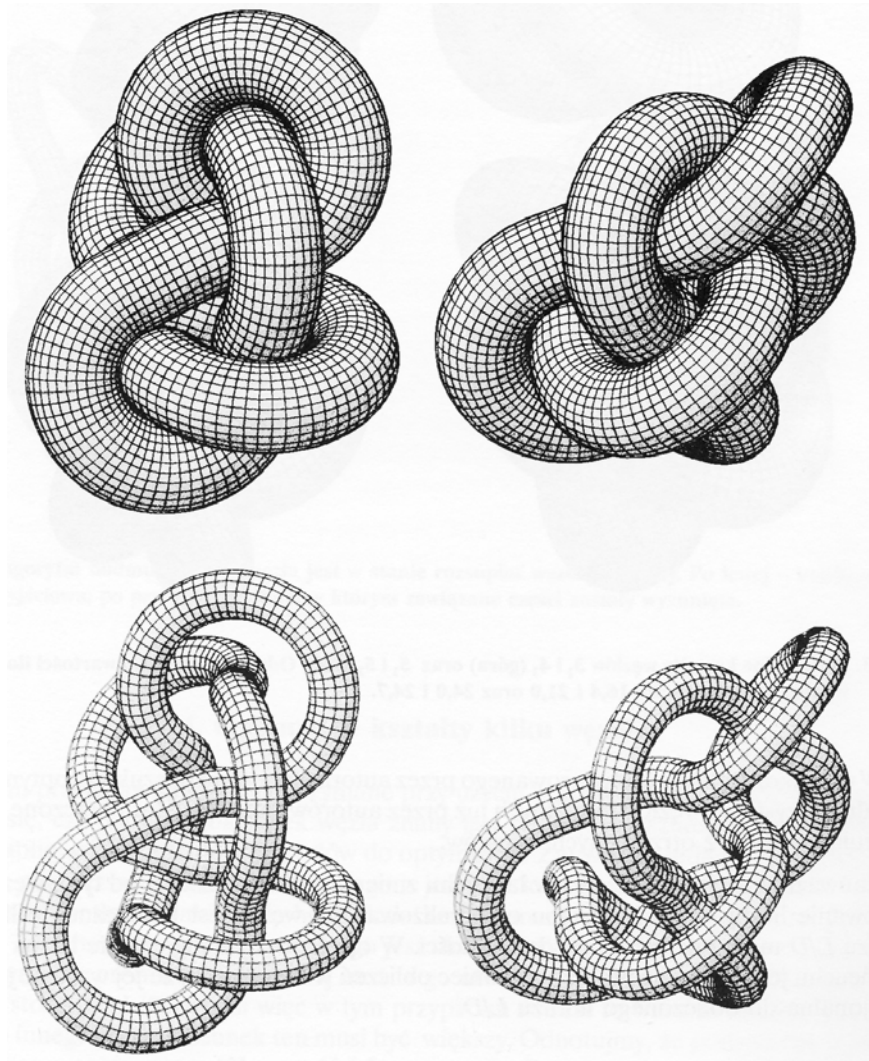
W celu skontrolowania opracowanego przez autora algorytmu poszukano optymalnego kształtu wszystkich węzłów zbadanych już przez autorów pracy [2]. Zamieszczone rysunki ilustrują niektóre z otrzymanych wyników.

Zauważmy, że liczba cząstek w łańcuchu zmienia się w zależności od typu węzła. Dostosowanie liczby cząstek do typu optymalizowanego węzła jest konieczne, jeśli wyniki ilorazu L/D mają mieć zbliżone dokładności. W opracowanym algorytmie liczba cząstek w łańcuchu jest normalizowana pod koniec obliczeń w ten sposób, że jej wartość jest proporcjonalna do obliczonego ilorazu L/D .

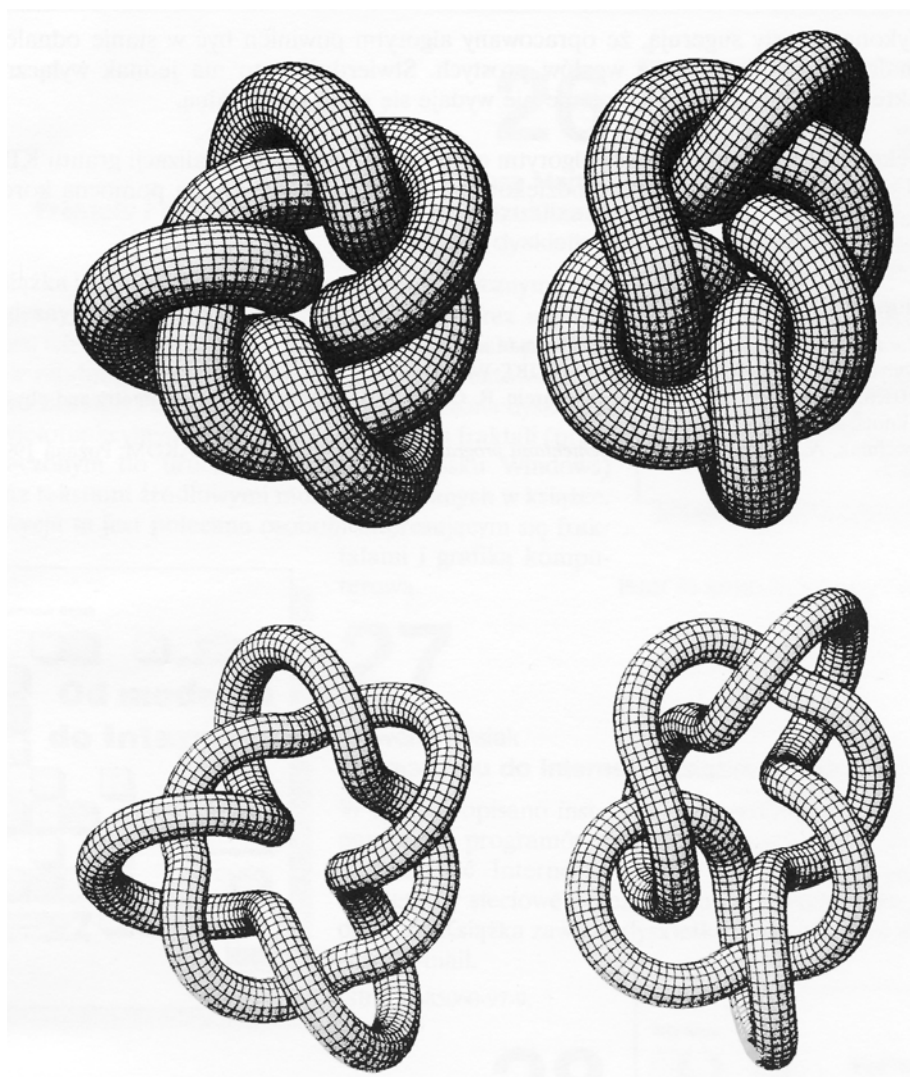
5. Wnioski

Przeprowadzone testy wykazują, że algorytm poszukiwania optymalnego kształtu węzłów przez symulację fizycznego procesu nadmuchiwanie rurki, z której węzły te wykonano-

no, jest szybki i skuteczny. Monitorowanie ewolucji kształtu optymalizowanego węzła ujawnia, że często w procesie wzrostu średnicy D pojawiają się spowolnienia, podczas których węzeł nadal zmienia swój kształt. Spowolnienia te prowadzą z reguły do nowej fazy ewolucji, podczas której ewolucja ulega gwałtownemu przyspieszeniu. Proces nadmuchiwania węzła wymaga kontroli eksperymentatora. W celu przyspieszenia ewolucji kształtu węzła może on na pewien czas zmniejszyć liczbę cząstek w łańcuchu lub zmienić kierunek



Rys. 4. Optymalne kształty węzłów 6_1 i 6_2 (góra). Wartość ilorazu L/D wynosi dla nich, odpowiednio 28,4 i 28,9. W celu rozjaśnienia sposobu, w jaki lina ułożona jest w przestrzeni, w dolnej części rysunku przerysowano optymalne formy obu węzłów z użyciem liny o przekroju o połowę mniejszym.



Rys. 5. Optymalne kształty węzłów 7_1 i 7_2 . Wartość ilorazu L/D wynosi dla nich, odpowiednio 30,8 i 32,7.

ich przepływu wywoływanego użyciem procedury SK. W ogólności, poszukiwanie optymalnego kształtu danego węzła przy wykorzystaniu opracowanego algorytmu przypomina wykonywanie realnego doświadczenia, w którym ingerencja eksperymentatora jest konieczna dla właściwego przebiegu eksperymentu. Na przykład, wybór zbyt małej liczby cząstek może doprowadzić do zerwania łańcucha i zmiany typu węzła. Graficzne monitorowanie procesu ewolucji pozwala uniknąć takiej katastrofy.

Wykonane testy sugerują, że opracowany algorytm powinien być w stanie odnaleźć optymalny kształt każdego z węzłów prostych. Stwierdzenie to ma jednak wyłącznie charakter hipotezy. Jej udowodnienie nie wydaje się sprawą trywialną.

Podziękowania. Opisany w pracy algorytm opracowano w ramach realizacji grantu KBN 8T11F 010 08P04. Autor składa podziękowania Panu A. Stasiakowi za pomocną korespondencję.

Literatura

- [1] Adams, C. C., *The Knot Book*, W. H. Freeman and Company, New York 1994.
- [2] Przytycki, J. H., *Węzły*, Wydawnictwo SCRIPT, Warszawa 1995.
- [3] Katrich, V., Bednar, J., Michoud, D., Scharein, R. G., Dubochet, J., Stasiak, A., Geometry and physics of knots, *Nature* 384 (1996), 142.
- [4] Marciniak, A., *Turbo Pascal 7.0 z elementami programowania*, Wydawnictwo NAKOM, Poznań 1994.