



СОФИЙСКИ УНИВЕРСИТЕТ “СВ. КЛИМЕНТ ОХРИДСКИ”  
ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

---

**Стохастичен анализ на количеството памет  
на алгоритмичната схема търсене с връщане**

---

Ангел Иванов Димитриев

*Научен ръководител:*  
гл. ас. Добромир Павлов Кралчев

ДИПЛОМНА РАБОТА

МАГИСТЪРСКА ПРОГРАМА  
“ЛОГИКА И АЛГОРИТМИ”

София, 2022 г.

## Съдържание

Увод . . . . .	2
1 Търсене на хамилтонов цикъл . . . . .	5
2 Задачата за цариците . . . . .	30
3 Судоку . . . . .	53
Заклучение . . . . .	60

## Увод

Информатиката разполага с множество алгоритмични задачи, в които се търси комбинаторна конфигурация със зададени свойства, но не е известен достатъчно бърз алгоритъм за нейното построяване. В такъв случай често използвана алтернатива на пълното изчерпване е алгоритмичната схема, позната под името търсене с връщане: обектът се строи стъпка по стъпка, като всеки елемент се избира така, че да удовлетвори изискванията по отношение на предишните елементи. Ако на някоя стъпка алгоритъмът не може да открие подходящ елемент, той се връща с една стъпка и поправя избора на предходния елемент. Ако и за предходния елемент в конфигурацията няма подходящ избор, алгоритъмът се връща с още една стъпка и т.н. Така броят на елементите в комбинаторния обект постоянно се променя: ту расте, ту намалява. В крайна сметка този брой или достига най-голямата възможна стойност (с което търсената комбинаторна конфигурация е построена / намерена), или достига нулата, изчерпвайки възможностите за първия елемент (което значи, че обектът не може да се построи, т.е. не съществува).

Важна характеристика на всеки алгоритъм е сложността по памет. Това е количеството допълнителна памет без паметта за входните данни, нужно за тяхната обработка. Текущото количество използвана памет се мени в хода на работата на алгоритъма, следователно стойностите му образуват времеви ред, който предлага разнообразни възможности за статистическа обработка: така например можем да се интересуваме от тенденцията на количеството памет (расте, намалява или се колебае около една определена стойност), от стандартното му отклонение или от средната стойност и т.н.

Сложността по памет зависи от входните данни на алгоритъма. Най-силно тя зависи от техния обем. Тъй като обаче при фиксиран обем има различни входни данни с такъв обем, в теорията се разглеждат различни видове сложност по памет: максимална, минимална и средна. Общото между всички тях е, че съответната математическа операция се извършва над множеството на всички входни данни с даден обем, а по времето за работа на алгоритъма традиционно се взема максимум. Последната презумпция изглежда основателна, поне на пръв поглед, защото паметта трябва да стигне за цялото изпълнение на алгоритъма, а не само за част от него.

На практика обаче е по-разумно да се използва друга мярка. Съвременните операционни системи са многозадачни и всяка програма се изпълнява едновременно с много други програми, споделяйки с тях общото налично количество памет на машината. Ако една програма се нуждае временно от повече памет, тя може да я вземе от общото количество памет на машината, предоставяйки я на другите програми, след като приключи работа с нея. Ето защо от практическа гледна точка не е проблем, ако една програма се нуждае от много памет временно, стига да използва малко памет през повечето време от своята работа. Ето защо е съществено не толкова максималното, колкото *типичното* количество памет, използвано от програмата по време на изпълнение. Под типично количество памет разбираме онази стойност, около която се колебае използваното количество памет през по-голямата част от времето за изпълнение на програмата. Изследванията показват, че при схемата търсене с връщане тази величина е добре дефинирана, в смисъл че наистина съществува определена стойност, около която се извършват колебания с малки отклонения през почти цялото време.

Типичното количество използвана памет зависи на първо място от *обема* на входните данни. То може да се различава съществено при различни входни данни с еднакъв обем. Често обаче се оказва, че тези различия могат да бъдат обхванати от втора характеристика, за която въвеждаме условното название *плътност* на входните данни; тук не предлагаме определение за плътност, защото нейният смисъл зависи от конкретната алгоритмична задача (въпреки че има прилики в начина на определяне при различни задачи). Нашите експерименти, извършени върху алгоритмичната схема търсене с връщане, показват, че при различни входни данни с еднакъв обем и еднаква плътност типичното количество памет зависи от конкретните входни данни съвсем слабо, което дава възможност за надеждно прогнозиране, а оттук и за планиране и избор на характеристиките на средата, в която ще се изпълнява дадена програма: общо количество памет, брой едновременно работещи програми, избор на операционна система и на стратегия за управление на паметта.

В дипломната работа създаваме теоретичен модел за промяната на количеството използвана памет по време на търсене с връщане. Моделът е приложим към всякакви алгоритми, основани на тази схема, прогнозира типичното количество памет, нужно на такива алгоритми, и обяснява различни закономерности, забелязани при експериментите.

Естественният избор е да мерим количеството памет в битове. Този избор не е твърде подходящ, защото поражда технически проблеми: налага се да знаем представянето на данните в паметта на компютъра, а то зависи както от програмата, така и от архитектурата на машината. Затова ще използваме друга, по-удобна мярка — текущия брой елементи на построявания комбинаторен обект. Ако например търсим пермутация с определени свойства, която трябва да се състои от общо  $n$  елемента и в някой момент сме задали стойностите на първите  $\xi$  от тях, ще смятаме, че количеството използвана памет в този момент е равно (по-точно, пропорционално) на  $\xi$ . Тъй като величината  $\xi$  се мени с изпълнението на алгоритъма, ще я бележим с  $\xi_t$ , където индексът  $t$  означава текущия момент. Алгоритъмът се изпълнява стъпка по стъпка, затова използваме модел с дискретно време:  $t \in \{0; 1; 2; 3; \dots; T\}$ , където  $T$  е броят на всички стъпки на алгоритъма. По определение  $\xi_t$  е броят на елементите на построявания, но възможно непълнен обект в момента  $t$ . С  $n$  ще бележим броя на елементите на пълния обект. Следователно  $0 \leq \xi_t \leq n$  за всяко  $t$ . При схемата търсене с връщане  $\xi_0 = 0$  за всякакви входни данни, а  $\xi_T = n$  за входни данни, при които търсеният обект съществува.

Макар че изследваме детерминирани алгоритми, можем да приемем, че входните данни  $\omega$  се избират случайно. При това тълкуване  $\xi_t = \xi_t(\omega)$  се превръща в случайна величина за произволно, но фиксирано  $t$ ; а когато  $t$  пробягва целите числа от 0 до  $T$ , се получава редицата  $(\xi_t)_{t \geq 0}$ , която представлява случаен процес. Неговите числови характеристики са основен интерес на настоящото изследване.

В дипломната работа се разглеждат подробно няколко задачи, от които става ясно как методът може да се прилага в по-обща ситуация. Глава 1 съдържа експериментален и теоретичен анализ на търсенето на хамилтонов цикъл в неориентиран граф. В глава 2 се предлага модел на търсенето с връщане в задачата за цариците върху шахматна дъска. Глава 3 е посветена на играта судоку. Заключениеето съдържа обобщение, описание на приносите на дипломната работа и някои възможни насоки за бъдещо развитие на изследванията в тази област.

# 1 Търсене на хамилтонов цикъл

Разглеждаме известната задача за търсене на хамилтонов цикъл в даден неориентиран граф с  $n$  върха. Без ограничение предполагаме, че върховете на графа са номерирани с целите числа от 1 до  $n$  включително. Представяме хамилтоновия цикъл  $v_1 - v_2 - v_3 - \dots - v_n - v_1$  като пермутация без повторение  $v_1, v_2, \dots, v_n$  на числата от 1 до  $n$ . Отново без ограничение можем да приемем, че  $v_1 = 1$ .

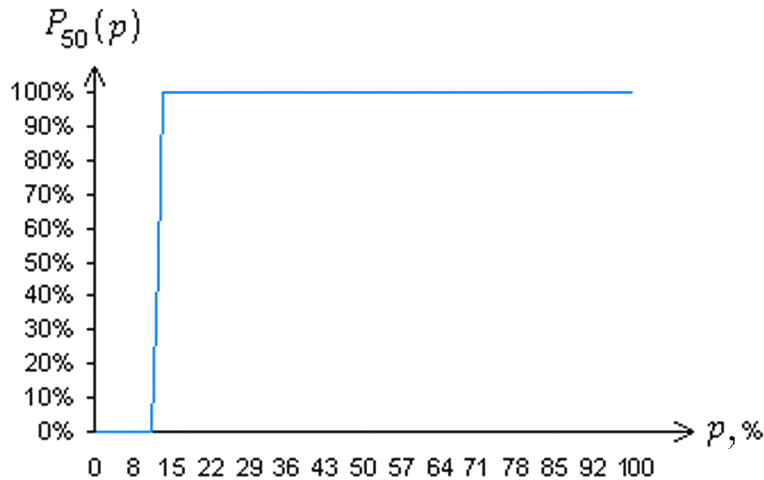
Търсенето с връщане протича така: първо търсим ребро  $\{v_1; v_2\}$ , за което  $v_1 \neq v_2$ . (Без ограничение смятаме, че графът няма примки, следователно всяко ребро  $\{v_1; v_2\}$  изпълнява изискването  $v_1 \neq v_2$ .) После търсим ребро  $\{v_2; v_3\}$ , за което  $v_3 \neq v_1$  и  $v_3 \neq v_2$ . Ако го намерим, търсим следващо ребро и тъй нататък. Нека вече сме намерили път  $v_1 - v_2 - v_3 - \dots - v_\xi$  и  $\xi < n$ . Търсим ребро  $\{v_\xi; v_{\xi+1}\}$ , за което  $v_{\xi+1} \neq v_1, v_2, \dots, v_\xi$ . Ако открием такова ребро, добавяме го към пътя:  $v_1 - v_2 - v_3 - \dots - v_\xi - v_{\xi+1}$ , след което увеличаваме  $\xi$  с единица. В противен случай премахваме от пътя реброто  $\{v_{\xi-1}; v_\xi\}$  и търсим друго ребро, излизащо от  $v_{\xi-1}$ . Ако не открием подходящо ребро, премахваме от пътя и реброто  $\{v_{\xi-2}; v_{\xi-1}\}$  и търсим друго ребро, излизащо от  $v_{\xi-2}$ . (Само за яснота бележим върховете  $v_{\xi-1}$  и  $v_{\xi-2}$  така. В действителност при всяко триене на ребро намаляваме  $\xi$  с единица, така че последният връх от пътя винаги да бъде отбелязан с  $v_\xi$ .) Търсенето завършва по един от следните начини:

- или изтриваме от пътя върха  $v_1$  и  $\xi$  става нула (това е признак, че не съществува хамилтонов цикъл);
- или намираме път  $v_1 - v_2 - v_3 - \dots - v_n$ , за който има ребро от  $v_n$  към  $v_1$  (с което е намерен хамилтонов цикъл).

Разглеждаме входните данни като случаен граф (вж. [1], глава VII): за всеки два различни върха вероятността да има ребро между тях е някакво реално число  $p \in [0; 1]$  — едно и също за всеки два върха; по определение тези случайни събития са независими в съвкупност.

И така, двата най-важни параметъра са броят на върховете  $n$  и вероятността  $p$  за съществуване на ребро между кои да е два върха;  $n \in \mathbb{N}$ ,  $n \geq 3$ ,  $p \in [0; 1]$ . Вероятността  $p$  наричаме плътност на графа. Означаваме с  $P_n(p)$  вероятността за съществуване на хамилтонов цикъл. Очевидно  $P_n(0) = 0$ ,  $P_n(1) = 1$  и  $P_n(p)$  е растяща функция на  $p$ .

Експериментите показват, че  $P_n(p)$  расте стръмно от 0 до 1. Например  $P_{50}(0,11) < 0,01$  и  $P_{50}(0,14) > 0,99$ , тоест почти цялото нарастване на  $P_n(p)$  от 0 до 1 се извършва, когато плътността  $p$  се мени от 0,11 до 0,14 — участък с ширина едва 0,03. Този участък се стеснява, т.е. наклонът на функцията става все по-стръмен, с увеличаването на  $n$ . По-точно, дължината на участъка на стръмно нарастване клони към 0 и  $P_n(p)$  се приближава към стъпаловидна функция на  $p$  при  $n \rightarrow \infty$ .



Тези наблюдения са в съгласие с теоремата на Поза [2] и уточнението на Коршунов [3], според които критичната стойност на плътността е

$$p_{\text{кр.}} \stackrel{\text{def}}{=} P_n^{-1}(0,5) \sim \frac{\ln n}{n}.$$

При  $n = 50$  формулата предсказва критична плътност  $p_{\text{кр.}} \approx 0,08$ , което е доста близо до критичния участък, получен експериментално. Формулата става все по-точна с нарастването на  $n$ .

Стойността на параметъра  $p$ , влияеща на генерирането на графа, е поначало неизвестна за алгоритъма, получил графа като входни данни. Но алгоритъмът знае броя  $n$  на върховете и с едно обхождане на графа (т.е. бързо — с линейна времева сложност) намира броя  $m$  на ребрата. Частното  $m/\binom{n}{2}$  (плътността на случайния граф) е неизместена оценка на параметъра  $p$ , затова алгоритъмът може да приеме, че  $p \approx m/\binom{n}{2}$ . Извън критичния участък можем да си спестим търсенето, приемайки, че вероятно има хамилтонов цикъл (за  $p$  надясно от критичния участък), съответно няма хамилтонов цикъл (за  $p$  наляво от критичния участък). Това е допустимо, когато искаме само да разпознаем вида на графа.

Надеждността на тези отговори зависи от точния начин на избиране на краищата на критичния участък. В примера по-горе избрахме краища  $P_{50}^{-1}(0, 01)$  и  $P_{50}^{-1}(0, 99)$ , което съответства на доверителна вероятност 99 % (това е надеждността на върнатата стойност на алгоритъма, когато  $p$  е извън критичния участък). Поради стръмния наклон на графиката изборът на доверителната вероятност влияе съвсем слабо на големината на критичния участък. Така или иначе, ясно е, че извън този участък прогнозата за вида на графа е достатъчно сигурна и можем да си спестим търсенето на хамилтонов цикъл, ако е важен само видът на графа (хамилтонов или нехамилтонов) и допускаме малка вероятност (0,01) за грешен отговор. При такава постановка (задача за разпознаване) е съществен само случаят, когато плътността  $p$  е около критичната. Ако държим да намерим хамилтонов цикъл (когато съществува такъв), пак плътностите около критичната се оказват най-важни за алгоритъма: при такива плътности търсенето е най-бавно, а в останалите случаи то бързо намира хамилтонов цикъл или установява, че няма такъв.

Очевидно прогнозата за изхода на търсенето е най-несигурна, когато  $p$  има такава стойност, че  $P_n(p) = 0,5$ . Тази конкретна плътност непременно принадлежи на критичния участък, независимо от избора на доверителната вероятност, определяща краищата му. Ето защо ще наричаме тази стойност на  $p$  критична плътност:  $p_{\text{кр.}} = P_n^{-1}(0,5)$ . Тъй като дължината на критичния участък клони към 0 при  $n \rightarrow \infty$ , то всичките му точки, вкл.  $p_{\text{кр.}}$ , имат еднакво асимптотично поведение, описано от формулата на Поза и Коршунов:

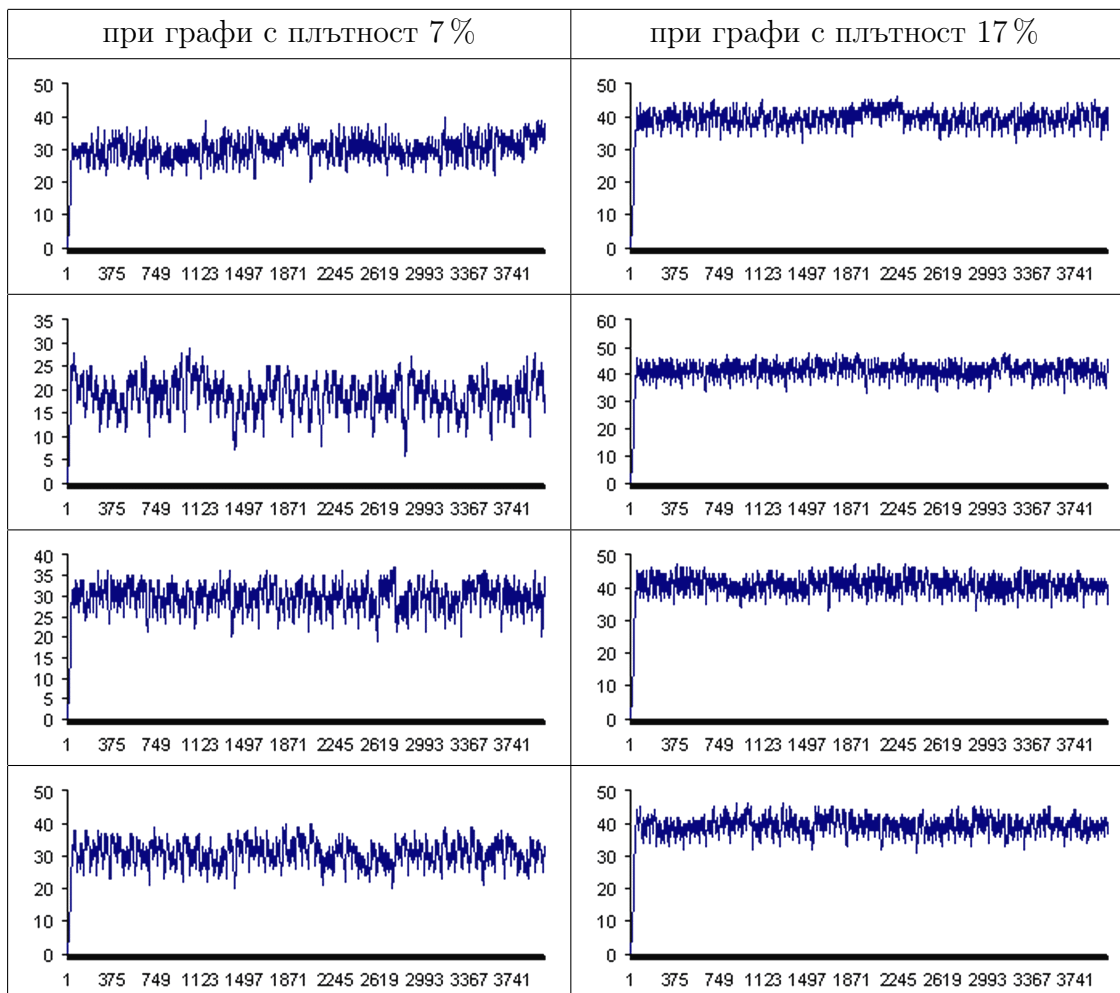
$$p_{\text{кр.}} \sim \frac{\ln n}{n}.$$

При плътности далече от критичната търсенето приключва бързо, поради което използваното количество памет не е от голямо значение: дори да се използва много памет, то ще е за кратко, после паметта ще се освободи за други програми. Но при графи с критична плътност търсенето работи бавно, ето защо е важно колко памет изразходва. Въпросът може да се изследва експериментално и теоретично.

Търсенето с връщане беше изпълнено върху графи с  $n = 50$  върха и плътности около критичната. При  $n = 50$  критичната плътност е 12 % (формулата на Поза и Коршунов дава стойност 8 %, но тази формула е вярна само асимптотично). За графи от този вид бяха наблюдавани траекториите на случайния процес  $(\xi_t)_{t \geq 0}$ , където  $\xi_t$  е броят на върховете в частта от търсения хамилтонов цикъл, построена към момента  $t$ .



Примерни траектории на случайния процес  $(\xi_t)_{t \geq 0}$ :



Лесно се забелязват и обясняват някои прилики на траекториите. Например повечето траектории са дълги, тоест времето  $T$  на алгоритъма е голямо (освен при специални входни данни). Причината е, че нашето търсене с връщане няма никакви допълнителни оптимизации. Ето защо времето  $T$  може да бъде от порядъка дори на факториел. Например да разгледаме граф, чиито върхове № 2, № 3, ..., №  $n-1$  образуват клика, връх № 1 е свързан само с № 2, а връх №  $n$  е свързан само с връх №  $n-1$ . За такъв граф търсенето с връщане е сходно с пълното изчерпване: обхождат се пермутациите на върховете от кликата, затова  $T \succeq (n-2)!$ .

Вижда се също, че всяка траектория се колебае около едно число (средна стойност на траекторията). С други думи, погледната отдалече, всяка траектория изглежда като водоравна права линия (с изключение на един стръмно растящ начален участък). Нулевият наклон на правата може да се обоснове лесно: дори при не много голям ненулев наклон правата скоро би излязла от допустимите стойности на ординатата. Наистина, по абсцисната ос е нанесено времето  $t$  на процеса,  $0 \leq t \leq T$ ; а по ординатната ос са нанесени стойностите  $\xi_t$  на процеса,  $0 \leq \xi_t \leq n$ . Тъй като отсечката от споменатата права започва почти от левия край и достига до десния край на правоъгълник  $T \times n$ , то наклонът на правата по абсолютна стойност не надхвърля  $n/T \approx n/(n-2)! \approx 0$  при големи  $n$ .

Стръмният начален участък също притежава просто обяснение. Тъй като  $\xi_0 = 0$  и  $0 \leq \xi_t \leq n$ , то отначало  $\xi_t$  може само да нараства. От  $\xi_t \leq n$  следва, че има не повече от  $n$  последователни нараствания в началото. Ето защо дължината на стръмния участък е  $O(n)$ , тоест той е къс в сравнение с останалата част от траекторията ( $n/T \approx 0$ ).

Нулевият наклон на дългия участък от траекториите оправдава въвеждането на понятието типично количество памет — това е числото, около което се колебае съответната траектория.

Дотук обръсахме внимание на приликите между траекториите. От значение обаче са и разликите между двете множества от траектории. Очевидно при плътност 17% траекториите се движат около числото 40; докато при плътност 7% те се колебаят на равнища около 25–30. В допълнение към графиките да разгледаме две извадки от равнища, около които се мени стойността на случайния процес (при  $n = 50$ ):

— При плътност 7% (почти всички графи са нехамилтонови):  
26,9; 25,3; 35,4; 32,4; 31,0; 21,8; 29,8; 31,9; 22,2.

Числови характеристики:

обем:  $n_x = 9$ ;

средноаритметично:  $\bar{x} = 28,5$ ;

средно отклонение:  $s_x = 4,7$ .

— При плътност 17% (почти всички графи са хамилтонови):  
44,0; 42,4; 37,8; 44,6; 41,1; 42,8; 40,8; 39,2; 39,2.

Числови характеристики:

обем:  $n_y = 9$ ;

средноаритметично:  $\bar{y} = 41,3$ ;

средно отклонение:  $s_y = 2,3$ .

Средните стойности на извадките се различават чувствително, а отклоненията — в по-малка степен. За да не основаваме изводите си на субективни преценки, ще проверим някои статистически хипотези [4].

Първо сравняваме отклоненията на двете генерални съвкупности. Основната хипотеза е, че са равни; избираме едностранна алтернатива, гласяща, че първото стандартно отклонение е по-голямо от второто. Хипотези за отклонения се проверяват чрез статистиката  $F$ :

$$F = \frac{\max(s_x^2, s_y^2)}{\min(s_x^2, s_y^2)} = \frac{4,7^2}{2,3^2} = 4,2.$$

На тази наблюдавана стойност съответства  $p$ -стойност 0,03, която е близо до стандартното ниво на доверие 0,05, но все пак е по-малка от него. Така че можем да отхвърлим основната хипотеза и да приемем, че отклоненията на двете групи от траектории са съществено различни. По-точно, отклоненията намаляват, когато плътността на графа расте. (Става дума за отклоненията между две траектории от една група, а не за отклоненията между стойностите от една траектория.)

Сега сравняваме средните стойности на генералните съвкупности. Основната хипотеза е, че са равни; избираме едностранна алтернатива, гласяща, че първата средна стойност е значително по-малка от втората. Тъй като отклоненията на генералните съвкупности са неизвестни, за проверката използваме статистиката  $T'$  с 12 степени на свобода:

$$T' = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{s_x^2}{n_x} + \frac{s_y^2}{n_y}}} = \frac{28,5 - 41,3}{\sqrt{\frac{4,7^2}{9} + \frac{2,3^2}{9}}} = -7,3.$$

На тази наблюдавана стойност съответства малка  $p$ -стойност: 0,000005; т.е. можем с почти пълна сигурност да приемем, че средното равнище, около което се колебаят траекториите, т.е. типичното количество памет, расте с увеличаване на плътността на графа.

Неформално казано, интервалите, в които се намират стойностите на двете генерални съвкупности, са добре отделени (не се смесват), поради което можем с голяма сигурност да разпознаем дадена извадка на коя генерална съвкупност принадлежи. Това става почти в началото на водоравния участък; така много скоро след началото на алгоритъма можем с голяма сигурност да определим типичното количество памет.

Например за графи с  $n = 50$  върха началният стръмен участък теоретично има дължина не повече от  $n = 50$ , както вече беше казано. Но разсъжденията предполагаха строга монотонност на траекторията. Практически такава липсва: стръмното изкачване съдържа спускания на отделни стъпки. За да сме сигурни, че стръмният участък е свършил, е добре да предположим, че дължината му е малко по-голяма от  $n$ . Експериментите показват  $2n = 100$  като приближение за дължината му. С други думи, събирането на данни за процеса  $\xi_t$  от водоравния участък е добре да започне от някой момент  $t \geq 2n$ . Запазва се оценката  $O(n)$  за дължината на стръмния участък, а обемът на извадката се определя по статистическите формули за планиране на обема, но априорно можем да допуснем ориентировъчно, че 30 наблюдения са достатъчни (извадки с обем  $\geq 30$  е прието в статистиката да се смятат за големи): по извадката  $(\xi_t)_{t=100}^{t=129}$  можем да определим типичното количество памет.

Разсъжденията дотук се основаваха на експериментални данни за графи с  $n = 50$  върха. За да се убедим, че този брой на върховете не е специален с нищо, ще анализираме случайния процес теоретично, когато броят  $n$  на върховете на графа е произволно цяло число,  $n \geq 3$ .

И така, правим стохастичен анализ на случайния процес  $(\xi_t)_{t \geq 0}$ . Нека в някой момент  $t$  е изпълнено равенството  $\xi_t = k$ , тоест в този миг текущо построената част от цикъла съдържа  $k$  върха и  $k - 1$  ребра.

От описанието на алгоритъма е ясно, че на следващата стъпка или добавяме един връх, или се отказваме от последния добавен връх; тоест  $\xi_{t+1} = \xi_t \pm 1$ , като знакът пред 1 е минус тогава и само тогава, когато не намираме следващ връх (това е връщането на алгоритъма). Означаваме с  $p_{n;k}$  вероятността на това събитие, тоест

$$p_{n;k} \stackrel{\text{def}}{=} \mathbf{P}\{\xi_{t+1} = \xi_t - 1 \mid \xi_t = k\} = \mathbf{P}\{\xi_{t+1} = k - 1 \mid \xi_t = k\}.$$

Лесно е да се изведе явна формула за тази вероятност. Ако текущият път (част от построявания хамилтонов цикъл) има вида

$$v_1 - v_2 - \dots - v_k,$$

то  $p_{n;k}$  е вероятността да няма ребро от  $v_k$  до никой от останалите  $n - k$  върха на графа. По правилото за умножение намираме

$$p_{n;k} = (1 - p)^{n-k}, \quad 1 \leq k \leq n.$$

Тъй като  $k = 0$  също е допустима стойност, полагаме  $p_{n;0} = 0$ , защото 0 е най-малкият възможен брой върхове в пътя (празен път).

Въпреки че  $0 \leq p \leq 1$ , граничните стойности не са интересни: при  $p = 1$  графът е почти сигурно пълен, затова има хамилтонов цикъл; при  $p = 0$  графът почти сигурно няма ребра и няма хамилтонов цикъл. Ето защо отгук нататък предполагаваме, че  $0 < p < 1$ .

При  $0 < p < 1$  изразът  $p_{n;k} = (1-p)^{n-k}$ ,  $1 \leq k \leq n$ , представлява строго растяща функция на  $k$ . Освен това  $p_{n;0} = 0$  и  $p_{n;n} = 1$ , тоест след нулева стойност случайният процес задължително нараства, а след стойност  $n$  задължително намалява.

Разсъжденията, с които изведохме формулата  $p_{n;k} = (1-p)^{n-k}$ , важат при  $1 \leq k < n$ . При  $k = n$  те водят до равенството  $p_{n;n} = 1$ , което изразява факта, че върховете на графа са изчерпани и няма как да добавим неизползвани върхове към пътя. Обаче това не е нужно. Щом търсим хамилтонов цикъл, трябва да проверим дали има ребро от последния към първия връх на пътя. Следователно  $p_{n;n} = 1 - p < 1$ . Ние обаче ще оставим в сила формулата  $p_{n;n} = 1$ , за да остане  $p_{n;k}$  растяща функция на  $k$ .

Както се вижда от графиките (може да се обоснове теоретично), достигане на някой от краищата 0 или  $n$  става след дълго време, а ние се интересуваме от поведението на процеса скоро след началото на водоравния участък от траекторията, затова можем да пренебрегнем екстремалните стойности  $k = 0$  и  $k = n$ .

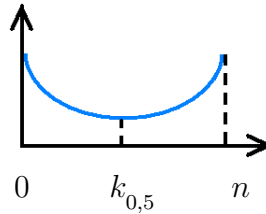
И така, имаме случаен процес с независими нараствания  $(\xi_t)_{t \geq 0}$ , който приема само стойностите  $0, 1, 2, \dots, n$ , като  $\xi_0 = 0$ ,  $\xi_{t+1} = \xi_t \pm 1$  и  $\mathbf{P}\{\xi_{t+1} = \xi_t - 1 \mid \xi_t = k\} = p_{n;k} = \begin{cases} 0, & k = 0; \\ (1-p)^{n-k}, & 1 \leq k \leq n. \end{cases}$

Тъй като  $p_{n;k}$  е непрекъсната функция на  $k$  и расте от 0 до 1, то за всяко  $\gamma \in (0; 1)$  има стойност на  $k$  (вкл. дробна), за която  $p_{n;k} = \gamma$ . Решаваме относно  $k$  уравнението  $p_{n;k} = \gamma$ , тоест  $(1-p)^{n-k} = \gamma$ , и намираме  $k = n - \log_{1-p} \gamma$ . Тази стойност на  $k$  означаваме с  $k_\gamma$ . Преобразуваме израза и получаваме

$$k_\gamma = n - \frac{\ln \gamma}{\ln(1-p)}.$$

Очевидно  $k_\gamma$  представлява строго растяща функция на  $\gamma$  и  $k_1 = n$ , както се очаква. Обаче  $k_\gamma$  клони към  $-\infty$  вместо към 0 при  $\gamma \rightarrow +0$ . Това разминаване съответства на уговорката, че  $p_{n;0}$  е 0, а не  $(1-p)^n$ . Всъщност  $k_\gamma = 0$  при  $\gamma = (1-p)^n$ . С други думи,  $k_\gamma$  се мени от 0 до  $n$ , когато  $\gamma$  пробягва интервала от  $(1-p)^n$  до 1, а не от 0 до 1. Разликата не е важна, тъй като  $(1-p)^n \rightarrow 0$  при  $n \rightarrow \infty$  и фиксирано  $p \in (0; 1)$ . Дори ако  $p$  не е фиксирано, а например  $p = \frac{\ln n}{n}$  (критичната плътност), все пак  $(1-p)^n = \left(1 - \frac{\ln n}{n}\right)^n \rightarrow 0$ .

Щом  $p_{n;k}$  е строго растяща функция на  $k$ , то  $p_{n;k} < 0,5$  за  $k < k_{0,5}$  и  $p_{n;k} > 0,5$  за  $k > k_{0,5}$ . Тоест за изменението на  $\xi$  важи правилото: ако  $\xi > k_{0,5}$ , то по-вероятно е  $\xi$  да намалее (и обратно).



Нагледно формулирано,  $k_{0,5}$  е точка на устойчиво равновесие, а вероятността  $p_{n;k}$  действа в ролята на връщаща сила: тя насочва  $\xi$  в посока към  $k_{0,5}$ . Стойностите на случайния процес се изменят като абсцисата на местоположението на топче, търкалящо се в купичка: водоравното дъно на купичката се явява точка на устойчиво равновесие. Отвесните краища на купичката съответстват на голяма връщаща сила, която пречи на топчето да излезе от купичката (краищата на интервала се достигат трудно, след многократни опити, тоест след дълго време).

В някакъв смисъл  $k_{0,5}$  е средна стойност на случайния процес. Средната стойност може обаче да се пресметне и по други съображения. Поради наличието на връщаща сила случайният процес се движи предимно в някаква околност на точката  $k_{0,5}$ , но не е задължително въпросната околност да е симетрична спрямо тази точка. Краищата на интервала съответстват на местата, в които връщащата сила става достатъчно голяма. При все че не сме уточнили коя сила смятаме за достатъчно голяма, логическата последователност изисква големината на тази сила да бъде една и съща в двата края. Затова трябва да изберем интервал от вида  $[k_{1-\gamma}; k_\gamma]$ ,  $0,5 < \gamma < 1$ .

Изборът на стойност на параметъра  $\gamma$  дава точен смисъл на думите “достатъчно голяма връщаща сила”. Независимо от конкретния избор интервалът винаги съдържа точката  $k_{0,5}$ . Но тя не е негова среда. Средата на интервала е точката

$$k_{\text{ср.}} = \frac{1}{2} (k_{1-\gamma} + k_{\gamma}),$$

за която след заместване от формулата за  $k_{\gamma}$  намираме

$$k_{\text{ср.}} = n - \frac{\ln(\gamma(1-\gamma))}{2 \ln(1-p)}.$$

Същевременно

$$k_{0,5} = n - \frac{\ln 0,5}{\ln(1-p)} = n + \frac{\ln 2}{\ln(1-p)}.$$

От неравенството  $\gamma(1-\gamma) < 0,25$ , което е в сила за всяко  $\gamma \in (0,5; 1)$ , следва, че  $k_{0,5} > k_{\text{ср.}}$ .

Кое от тези числа е средноаритметично (математическо очакване) на стойностите на случайния процес от водоравния участък? Оказва се, че това е  $k_{\text{ср.}}$ . Понеже  $p_{n;k} \approx 0,5$  за  $k \approx k_{0,5}$ , то в околност на точката  $k_{0,5}$  случайният процес има характер на симетрично случайно лутане [5], ограничено във въпросната околност, която можем да отъждествим приблизително с интервала  $[k_{1-\gamma}; k_{\gamma}]$ , закръглявайки краищата му до цели числа. Това лутане може да се представи чрез верига на Марков, която е:

- крайна, защото интервалът съдържа краен брой цели числа;
- еднородна, защото вероятностите за преход не се менят с времето;
- неразложима, защото всяка стойност е достижима от всяка друга.

При движение със стъпка плюс-минус единица текущата стойност всеки път сменя четността си, затова разглежданата верига на Марков не е апериодична. Тя има два циклични подкласа, съставени съответно от четните и нечетните числа в интервала  $[k_{1-\gamma}; k_{\gamma}]$ . Всеки подклас поражда отделна верига на Марков, която е не само крайна, еднородна и неразложима, но също и апериодична. Щом има тези четири свойства, то всяка от новите вериги е ергодична, следователно притежава гранично и единствено стационарно разпределение и те съвпадат [5].

Нека  $\pi$  е граничното и стационарно разпределение на коя да е от двете вериги на Марков (четните и нечетните числа в интервала). Щом  $\pi$  е гранично разпределение, то се състои от вероятностите на стойностите на нашия случаен процес след достатъчно дълго време (т.е. след достигане на началото на водоравния участък от траекторията на случайния процес). А тъй като  $\pi$  е стационарно разпределение,  $\pi_k = \frac{1}{2}\pi_k + \frac{1}{4}\pi_{k-2} + \frac{1}{4}\pi_{k+2}$  за всяко цяло  $k$  от вътрешността на интервала ( $1/2, 1/4$  и  $1/4$  са вероятностите за нараствания  $0, +2$  и  $-2$  съответно). От това рекурентно уравнение следва, че

$$\pi_k = \frac{1}{2}\pi_{k-2} + \frac{1}{2}\pi_{k+2},$$

тоест редицата  $\pi$  е аритметична прогресия. Щом лутането е симетрично, вероятностите на краищата на интервала (първият и последният член на прогресията) са равни. Затова разликата ѝ е нула, откъдето следва, че всички членове са равни. Следователно  $\pi$  е равномерно разпределение и математическото му очакване съвпада със средата  $k_{\text{ср.}}$  на интервала (с абсолютна грешка, ненадхвърляща  $1/2$ ).

И така,

$$k_{\text{ср.}} = n - \frac{\ln(\gamma(1-\gamma))}{2 \ln(1-p)}$$

е типичното количество използвана памет — равнището, около което се колебаят стойностите на случайния процес във водоравния участък.

Стойността на параметъра  $\gamma$  в тази формула се избира така, че симетричното случайно лутане, ограничено в интервала  $[k_{1-\gamma}; k_\gamma]$ , да наподобява възможно най-добре случайния процес  $(\xi_t)_{t \geq 0}$  в  $[0; n]$ . Критерия за близост между процесите ще поясним със следния пример. Нека текущата стойност на  $\xi_t$  е някакво  $k$ , за което  $p_{n;k} = 1/8$ , тоест има вероятност  $1/8$  за стъпка наляво. Това не е симетрично лутане: при него тази вероятност е  $1/2$ . Но при симетрично случайно лутане вероятността за три последователни стъпки наляво е равна също на  $1/8$ . Значи три стъпки на симетричното лутане съответстват на една стъпка на процеса  $(\xi_t)_{t \geq 0}$ , чиито стойности се ограничават от връщащата сила, а симетричното лутане за компенсация се ограничават изкуствено в интервала  $[k_{1-\gamma}; k_\gamma]$ .



В общия случай дефинираме един мащабен множител  $\mu = \mu(k)$ :

$$p_{n;k} = \left(\frac{1}{2}\right)^\mu.$$

Неговият смисъл е, че ако процесът  $(\xi_t)_{t \geq 0}$  извърши стъпка наляво от точката  $k$ , това съответства (по вероятност) на  $\mu(k)$  стъпки наляво, ако процесът би представлявал симетрично случайно лутане. Очевидно

$$\mu = \mu(k) = -\log_2 p_{n;k}.$$

Тъй като  $\xi_t$  се мени в интервала от 0 до  $n$ , съдържащ  $n$  единици, а пък симетричното случайно лутане, с което го приближаваме, е ограничено в интервала от  $k_{1-\gamma}$  до  $k_\gamma$ , то трябва да важи равенството

$$\sum_{k=k_{1-\gamma}+1}^{k_\gamma} \mu(k) = n, \text{ тоест } \sum_{k=k_{1-\gamma}+1}^{k_\gamma} \log_2 p_{n;k} = -n.$$

Това равенство е критерият за близост между двата процеса. От него като от уравнение намираме  $\gamma$ :

$$\begin{aligned} \sum_{k=k_{1-\gamma}+1}^{k_\gamma} \log_2(1-p)^{n-k} = -n &\iff \sum_{k=k_{1-\gamma}+1}^{k_\gamma} (n-k) \log_2(1-p) = -n \\ \iff \frac{\ln(1-p)}{\ln 2} \sum_{k=k_{1-\gamma}+1}^{k_\gamma} (n-k) = -n &\iff \sum_{k=k_{1-\gamma}+1}^{k_\gamma} (n-k) = -\frac{n \cdot \ln 2}{\ln(1-p)} \\ \iff \left[ (n-k_\gamma) + (n-k_{1-\gamma}) \right] (k_\gamma - k_{1-\gamma}) &= -\frac{2n \cdot \ln 2}{\ln(1-p)} \iff \\ \frac{\ln(1-\gamma) + \ln \gamma}{\ln(1-p)} \cdot \frac{\ln(1-\gamma) - \ln \gamma}{\ln(1-p)} &= -\frac{2n \cdot \ln 2}{\ln(1-p)} \iff \\ \ln^2(1-\gamma) - \ln^2 \gamma &= -2n \cdot \ln 2 \cdot \ln(1-p). \end{aligned}$$

Ако  $n \rightarrow \infty$ , а  $p$  е фиксирано или е равно на критичната плътност, то дясната страна клони към безкрайност, затова  $\gamma \rightarrow 1$ ,  $\ln \gamma \rightarrow 0$  и

$$\ln^2(1-\gamma) \approx -2n \cdot \ln 2 \cdot \ln(1-p),$$

откъдето намираме

$$\gamma \approx 1 - \exp\left(-\sqrt{-2n \cdot \ln 2 \cdot \ln(1-p)}\right).$$

Това е подходящата стойност за замяната на единия процес с другия.

Във формулата  $k_{\text{ср.}} = n - \frac{\ln(\gamma(1-\gamma))}{2 \ln(1-p)}$  можем да заместим  $\gamma$ ;

но по причини, които ще изясним след малко, е по-добре да отделим зависимостта на  $k_{\text{ср.}}$  от  $\gamma$ . За целта записваме формулата във вида

$$k_{\text{ср.}} = n + \frac{\text{critParam}}{\ln(1-p)}, \quad \text{където } \text{critParam} = -\frac{1}{2} \ln(\gamma(1-\gamma)).$$

Предстои да изпробваме формулите чрез експерименталните данни за графи с  $n = 50$  върха, които данни бяха цитирани по-горе.

При критичната плътност  $p \approx 12\%$ :  $\gamma \approx 0,949$ ,  $\text{critParam} \approx 1,5145$ . Ако закръглим  $\gamma$  на  $0,95$ , то  $\text{critParam} \approx 1,52351$ .

За графи с плътност  $p = 7\%$  имаме  $\gamma \approx 0,894$ ,  $\text{critParam} \approx 1,178$ ,  $k_{\text{ср.}} \approx 33,8$ , което е сравнително близо до експериментално намерената средна стойност  $\bar{x} = 28,5$ .

За графи с плътност  $p = 17\%$  имаме  $\gamma \approx 0,9725$ ,  $\text{critParam} \approx 1,811$ ,  $k_{\text{ср.}} \approx 40,3$ , което е значително по-близо до експериментално намерената средна стойност  $\bar{y} = 41,3$ .

Любопитна подробност е, че изборът на  $\text{critParam} = 1,52351$  (съответстващо на критичната плътност) дава по-точни приближения дори за плътности, различни от критичната. Например при  $p = 7\%$  се получава  $k_{\text{ср.}} \approx 29,0$ , което лежи изключително близо до  $\bar{x} = 28,5$ . При  $p = 17\%$  се получава  $k_{\text{ср.}} \approx 41,8$ , което е много близо до  $\bar{y} = 41,3$ . Възможно обяснение е, че независимо от плътността на графа като цяло (която вероятно влияе на  $k_{\text{ср.}}$  посредством стръмния начален участък от траекторията на процеса) колебанията във водоравния участък най-често се извършват в подграф с локална плътност около критичната, защото плътностите, различни от критичната, водят не до колебания, а до бързо завършване на алгоритъма. Локалната плътност на подграфа е определяща за случайното лутане, а оттам също за  $\gamma$  и  $\text{critParam}$ . Дали това обяснение е правилно, ще може да се разбере със сигурност само след нови експерименти — възможна тема за бъдещо изследване.

Точката на устойчиво равновесие, макар и важна сама по себе си, се различава от стойността, около която се колебае траекторията във водоравния участък, както беше предвидено от нашия модел. При  $p = 7\%$  и при  $p = 17\%$  съответните  $k_{0,5} = 40,4$  и  $k_{0,5} = 46,3$  се намират далече от  $\bar{x} = 28,5$  и  $\bar{y} = 41,3$ .

Опростяваме получените изрази за плътности около критичната:

$$p \approx p_{\text{кр.}} \sim \frac{\ln n}{n}.$$

Заместваме в израза за  $k_{0,5}$  и взимаме предвид, че  $p \rightarrow 0$  при  $n \rightarrow \infty$ , следователно  $\ln(1-p) \sim -p$ .

$$k_{0,5} = n + \frac{\ln 2}{\ln(1-p)} \approx n + \frac{\ln 2}{-p} \approx n - \frac{\ln 2}{(\ln n)/n} = n \left(1 - \frac{\ln 2}{\ln n}\right).$$

От формулата за смяна на основата на логаритъма следва, че

$$k_{0,5} \approx n \left(1 - \frac{1}{\log_2 n}\right).$$

Това е точката на устойчиво равновесие, която привлича стойностите на случайния процес при графи с критична плътност.

Аналогично,

$$\gamma \approx 1 - \exp\left(-\sqrt{2 \cdot \ln 2 \cdot \ln n}\right),$$

$$\text{critParam} = -\frac{1}{2} \ln(\gamma(1-\gamma)) \approx \sqrt{0,5 \cdot \ln 2 \cdot \ln n},$$

$$k_{\text{кр.}} \approx n \left(1 - \frac{\text{critParam}}{\ln n}\right)$$

е математическото очакване на процеса във водоравния участък, тоест типичното количество памет, при графи с критична плътност. Асимптотични оценки:  $k_{\text{кр.}} \sim n = \Theta(n)$ ,  $k_{0,5} \sim n = \Theta(n)$ .

За  $n = 50$  тези формули прогнозираят следното:  $k_{0,5} \approx 41$ ;  $\gamma \approx 0,90$ ;  $\text{critParam} \approx 1,22$  по формулата с  $\gamma$  и  $1,16$  по формулата с радикала;  $k_{\text{кр.}} \approx 34,5$  и  $35,1$  съответно. Без опростяванията тези величини имат следните стойности  $\gamma \approx 0,95$ ;  $\text{critParam} \approx 1,52351$ ;  $k_{\text{кр.}} \approx 30,5$ .

Опростените формули са по-неточни от първоначалните формули, тъй като в допълнение към предишните приближения разчитат също на формулата на Поза и Коршунов, а тя е само асимптотично точна. Например за  $n = 50$  тя предвижда критична плътност малко под 8%, докато истинската стойност е около 12%.

Важна е не само средната стойност, но и отклоненията от нея. Приближение на стандартното отклонение  $\sigma$  може да бъде намерено чрез същия метод — замяна на изследвания процес със случайно лутане.

Това, че  $p_{n;k} \approx 0,5$  за  $k \approx k_{0,5}$ , означава, че близо до точката  $k_{0,5}$  практически не действа връщаща сила. Тя започва да се чувства едва когато процесът се отдалечи от точката  $k_{0,5}$ . Казано по-формално, близо до  $k_{0,5}$  процесът има характер на симетрично случайно лутане. Нека  $\eta_0 = \frac{n}{2}$ ,  $\eta_{t+1} = \eta_t \pm \frac{1}{2}$ ,  $\mathbf{P}\{\eta_{t+1} = \eta_t + \frac{1}{2}\} = \mathbf{P}\{\eta_{t+1} = \eta_t - \frac{1}{2}\} = \frac{1}{2}$  и нарастванията  $\eta_{t+1} - \eta_t$  са независими. Тогава стойността на  $\eta_t$  е някое измежду следните  $t + 1$  числа:  $\frac{n-t}{2}, \frac{n-t}{2} + 1, \frac{n-t}{2} + 2, \dots, \frac{n+t}{2}$ ; тоест симетричното случайно лутане е неограничено в пространството и отклонението му расте неограничено:  $\sigma(\eta_t) \rightarrow +\infty$  при  $n \rightarrow \infty$ . А стойностите на нашия случаен процес са ограничени между 0 и  $n$  вкл. Следователно отклонението му расте ограничено:  $\exists \sigma = \lim_{t \rightarrow \infty} \sigma(\xi_t)$ . За големи  $t$  важи приближението  $\sigma(\xi_t) \approx \sigma$  и случайната величина  $\xi_t$  приема някоя от следните  $n + 1$  стойности: 0, 1, 2, 3,  $\dots$ ,  $n$ . Тъй като  $\sigma(\eta_t) \rightarrow \infty$ , то процесът  $\eta$  само в първите си стъпки (в околност на  $n/2$ ) моделира процеса  $\xi$  в околност на  $k_{0,5}$ . Граничния момент откриваме чрез броя на възможните стойности:  $n + 1$  стойности има величината  $\eta_n$  и те са тъкмо числата 0, 1, 2, 3,  $\dots$ ,  $n$ . Ето защо можем да приемем, че  $\eta_n$  и  $\xi_t$  при  $t \rightarrow \infty$  са приблизително еднакво разпределени величини. Следователно

$$\sigma = \lim_{t \rightarrow \infty} \sigma(\xi_t) \approx \sigma(\eta_n) = \sqrt{n \cdot \frac{1}{2} \cdot \left(1 - \frac{1}{2}\right)} = \frac{1}{2}\sqrt{n}.$$

Равенството пред първия корен произтича от това, че  $\eta_n \in \text{Vi}(n; \frac{1}{2})$ , където  $\text{Vi}$  е обозначението за биомно разпределение [6].

И така, известно време след началото си случайният процес  $(\xi_t)_{t \geq 0}$  започва да се колебае около една средна стойност (изследвана по-горе) със стандартно отклонение

$$\sigma \approx \frac{1}{2}\sqrt{n}.$$

Това приближение е вярно само по порядък, защото е получено чрез замяна на един процес с друг. Освен това случайният процес  $\xi$  може да се смята за симетрично случайно лутане само в околност на  $k_{0,5}$ , а не в целия интервал от 0 до  $n$ , чиито краища посещава твърде рядко поради връщащата сила. Ето защо получената оценка за  $\sigma$  е завишена.

Изведената формула за  $\sigma$  зависи единствено от  $n$ . При  $n = 50$  получаваме приближението  $\sigma \approx 3,5$  (важи за всякакви плътности). А с помощта на извадки от експерименталните данни по-горе намерихме  $s_x = 4,7$  и  $s_y = 2,3$ , като установихме чрез статистиката  $F$ , че има значима разлика между отклоненията на двете генерални съвкупности. Приближението има голяма относителна грешка: стойностите на  $s_x$  и  $s_y$  се различават от  $\sigma$  с 1,2, което е цели 34% от стойността на  $\sigma$ . Разминаването в изводите и стойностите се дължи на различния смисъл на величините:  $s_x$  и  $s_y$  са отклонения на средните стойности на траекториите от тяхното средноаритметично (тоест от средното на средните). Обратно,  $\sigma$  измерва отклонението на стойностите на една траектория от нейната средна стойност. Ето защо не бива да сравняваме  $\sigma$  с  $s_x$  и  $s_y$ . Вместо това трябва да вземем по една извадка от всеки вид траектории и да пресметнем отклоненията на стойностите на всяка траектория поотделно.

Следните две извадки от експерименталните данни съдържат типични средноквадратични отклонения на стойности в една траектория от нейната средна стойност (взети са по шест траектории от всеки вид):

- за графи с плътност 7%: 2,9; 2,6; 2,1; 1,2; 2,5; 2,3;
- за графи с плътност 17%: 1,7; 2,4; 2,2; 1,9; 1,5; 2,1.

Средноаритметичните на двете извадки са съответно 2,3 и 2,0. Вижда се, че те се различават незначително. Това показва и проверката с теста на Стюдънт: получава се голяма р-стойност (приблизително 0,3). Този извод (за равенството на отклоненията) подкрепя разсъжденията от теоретичния анализ:  $\sigma$  не зависи от плътността на графа.

Обаче относителната грешка на формулата за  $\sigma$  е голяма — около четири десети от стойността на  $\sigma$ . Ето защо тази формула е вярна само по порядък. Както беше предвидено в теоретичния анализ, тя дава завишена оценка на отклонението.

Забелязаните закономерности позволяват съществено оптимизиране на разпознаването дали даден неориентиран граф е хамилтонов, или не. Ако графът е хамилтонов, ще бъде достатъчно да установим този факт, без да намираме конкретен хамилтонов цикъл. Идеята на оптимизацията е, че можем да пуснем търсене с връщане и да го прекратим много рано, като използваме събраните данни за разпознаване на вида на графа.

Тъй като вероятността за съществуване на хамилтонов цикъл и математическото очакване на типичното количество използвана памет са строго растящи функции на плътността  $p$  на графа при произволен, но фиксиран брой  $n$  на върховете му, то следва, че вероятността за съществуване на хамилтонов цикъл е строго растяща функция на математическото очакване на типичното количество използвана памет. Неизместена оценка на въпросното математическо очакване се получава от средноаритметичната стойност на случайния процес във водоравния участък на траекторията.

По-горе установихме, че споменатото математическо очакване

$$k_{\text{cp.}} \approx n \left( 1 - \frac{\text{critParam}}{\ln n} \right)$$

за графи с критична плътност. Оттук произтича следният алгоритъм за разпознаване на хамилтонови графи:

- 1) Пускаме стандартния алгоритъм за търсене с връщане, като следим стойностите на случайния процес  $\xi_t$ .
- 2) Щом започне водоравният участък на траекторията, образуваме извадка от достатъчно стойности на процеса.
- 3) Пресмятаме средното аритметично  $\bar{\xi}$  на извадката.
- 4) Ако  $\bar{\xi} > k_{\text{cp.}}$ , приемаме, че графът е хамилтонов; в противен случай приемаме, че графът не е хамилтонов.

Разпознаването на началото на водоравния участък може да стане по различни начини: с проверка на хипотези за знаците на нарастванията или просто чрез изчакване на предварително определено време. Както обяснихме по-горе, дължината на стръмния начален участък е от порядък  $O(n)$ ; числовият множител пред порядъка е между 1 и 2 според експерименталните данни. За да застраховаме алгоритъма срещу попадане в този участък, е добре да вземем горната граница 2.

Колко голяма извадка трябва да се образува, можем да намерим по формулата за планиране на обема на извадката [4]:

$$n' = \left\lceil \left( (Z_{1-\alpha} + Z_{1-\beta}) \cdot \frac{\sigma}{d} \right)^2 \right\rceil.$$

Тук  $\alpha$  и  $\beta$  са вероятности — отнапред избрани числа между 0 и 1. А числата  $Z$  са квантили на стандартното нормално разпределение.

Числата  $\alpha$  и  $\beta$  са вероятностите за грешки от първи и втори род (двата възможни грешни отговора на алгоритъма). По-малки  $\alpha$  и  $\beta$  водят до по-голяма извадка, по-сигурен анализ, но по-дълга обработка. Тези числа се избират от потребителя в разумни граници и се подават на входа на алгоритъма като негови параметри. Не зависят от броя  $n$  на върховете на графа, затова квантилите са константи спрямо  $n$ .

Числото  $\sigma$  е стандартното отклонение на стойностите на  $\xi_t$ , взети от водоравния участък. Можем да използваме оценката  $\sigma \approx 0,5\sqrt{n}$ . Знаем, че тя е завишена и е вярна само по порядък, но това не е пречка: от една страна, по-голямо  $\sigma$  води до по-голяма извадка, следователно по-надежден анализ; от друга страна, интересуваме се от горна граница за обема на извадката.

Числото  $d$  се нарича допустимо отклонение или значима разлика. Може да бъде избрано като някаква част от стандартното отклонение (конкретната част е параметър, задаван от потребителя). В този случай  $d = \Theta(\sqrt{n})$ ,  $d/\sigma = \Theta(1)$  и  $n'$  е константа спрямо  $n$ :  $n' = \Theta(1)$ .

Тъй като се интересуваме от горна граница за обема на извадката, трябва да вземем колкото може по-малко  $d$ . Но стойностите на процеса са цели числа, поради което смисленият минимум за  $d$  е  $d = 1$ . На практика  $d$  не бива да се избира така; този избор служи само за извеждане на горна граница. Във формулата за обема на извадката заместваме  $d = 1$  и  $\sigma = 0,5\sqrt{n}$ , при което получаваме обем

$$n' = \left\lceil \left( (Z_{1-\alpha} + Z_{1-\beta}) \cdot \frac{0,5\sqrt{n}}{1} \right)^2 \right\rceil = \lceil 0,25 (Z_{1-\alpha} + Z_{1-\beta})^2 n \rceil = \Theta(n).$$

Това е горна граница, т.е. извадката има обем  $n' = O(n)$ . Толкова стъпки продължава вторият етап — събирането на данни за процеса.

За определеност да изберем например  $\alpha = \beta = 0,025$ . Следователно  $Z_{1-\alpha} = Z_{1-\beta} = Z_{0,975} = 1,96 \approx 2$  и  $n' \approx n$ . Ето защо е удобно да работим с извадка с обем  $n' = n$ . В такъв случай

$$\bar{\xi} = \frac{\xi_{2n+1} + \xi_{2n+2} + \dots + \xi_{3n}}{n}.$$

При този избор на управляващите параметри алгоритъмът разпознава даден граф с  $n$  върха като хамилтонов тогава и само тогава, когато

$$\frac{\xi_{2n+1} + \xi_{2n+2} + \dots + \xi_{3n}}{n} > n \left( 1 - \frac{\text{critParam}}{\ln n} \right).$$

Времето за изчисления по тази формула е  $\Theta(n)$ , измервано с броя на аритметичните действия.

Събирането на данните преди това изисква  $3n = \Theta(n)$  стъпки, всяка от които представлява обхождане на списъка на ребрата, излизащи от текущия връх. Броят им е най-много  $n - 1 = O(n)$ , затова времето на този етап от алгоритъма е  $\Theta(n) \cdot O(n) = O(n^2)$ .

Времето на целия алгоритъм е сборът от времената на двете фази:  $O(n^2) + \Theta(n) = O(n^2)$  при всякакви входни данни. За плътни графи тази сложност е линейна, тъй като дължината на входа е  $\Theta(n^2)$  дори при най-икономично представяне — чрез списъци на съседствата.

Например при графи с петдесет върха и критична плътност (12%) неоптимизираното търсене с връщане често продължава много дълго, поне трийсет хиляди стъпки. Оптимизираният алгоритъм дава отговор след не повече от сто и петдесет стъпки, тоест поне двеста пъти по-бързо. За произволно  $n$  времевата сложност при най-лоши входни данни намалява, благодарение на оптимизацията, от  $\Omega((n - 2)!)$  на  $O(n^2)$ .

И така, описаната оптимизация значително ускорява алгоритъма (без нея времето е от порядъка на факториел при лоши входни данни, както беше показано по-горе). За сметка на това ускорение обаче алгоритъмът може да сбърка при разпознаването на някои графи. Възможни са два типа грешки:

- тип № 1: хамилтонов граф да бъде класифициран като нехамилтонов;
- тип № 2: нехамилтонов граф да бъде класифициран като хамилтонов.

Не е трудно да подведем алгоритъма с нарочно подбрани входни данни. Например два върха от първа степен, свързани към клика от  $n - 2$  върха, образуват граф, в който има хамилтонов път, но не и хамилтонов цикъл. Затова процесът ще се колебае около висока стойност (близка до  $n$ ) и алгоритъмът ще приеме графа за хамилтонов, въпреки че не е такъв. Само че случайното получаване на граф от този вид е малко вероятно: при голяма плътност не е вероятно да има върхове от първа степен; а при малка плътност не е вероятно образуването на голяма клика. Тоест оптимизираният алгоритъм е най-подходящ за случайни графи (например за подготвяне на тестови данни за някакви други алгоритми). Предложеният теоретичен модел се отнася именно за случайни графи. От направения стохастичен анализ става ясно, че при такива графи вероятността за грешка е малка. Точно колко често греша алгоритъмът, най-сигурно се установява с помощта на подходящи тестови данни. Алгоритъмът беше програмиран на C++ за целите на тестването.



```

#include <iostream>
#include <algorithm>
#include <vector>
#include <math.h>
#include <numeric>
#include <string>
#include <functional>
#include <sstream>
#include <fstream>

using namespace std;

class Graph
{
    size_t n;
    size_t m;
    std::vector<std::vector<size_t>> adj;
public:
    Graph(size_t n) : n(n), m(0), adj(n) {}
    void addEdge(size_t start, size_t end)
    {
        adj.at(start).push_back(end);
        adj.at(end).push_back(start);
        m++;
    }
    const std::vector<size_t>& getAdjVertices(size_t vertex) const
    {
        return adj.at(vertex);
    }
    size_t getVerticesCount() const
    {
        return n;
    }
    size_t getEdgesCount() const
    {
        return m;
    }
};

```

```

int backtrackingHamCycle(const Graph& currentGraph,
    size_t start, size_t currentVertex, std::vector<bool>& visited,
    size_t recursionDepth, size_t& time, std::vector<size_t>& recData,
    size_t firstPhaseEnd, size_t secondPhaseEnd)
{
    #define ADVANCE_TIME \
    \
    time++; \
    \
    if (time > firstPhaseEnd && time <= secondPhaseEnd) \
        recData.push_back(recursionDepth); \
    \
    if (time > secondPhaseEnd) \
        return -1; // Data collected; graph type unknown.

    recursionDepth++;
    ADVANCE_TIME
    visited[currentVertex] = true;
    auto currentAdj = currentGraph.getAdjVertices(currentVertex);
    for (size_t i = 0; i < currentAdj.size(); i++)
    {
        size_t neighbour = currentAdj[i];
        if (neighbour == start &&
            recursionDepth == currentGraph.getVerticesCount()
        ) return +1; // Hamiltonian cycle found.
        if (visited[neighbour]) continue;
        int res = backtrackingHamCycle(
            currentGraph, start, neighbour, visited, recursionDepth,
            time, recData, firstPhaseEnd, secondPhaseEnd);
        if (res != 0) return res;
        ADVANCE_TIME
    }
    visited[currentVertex] = false;
    recursionDepth--;
    return 0; // Hamiltonian cycle not found.

    #undef ADVANCE_TIME
};

```

```

bool isHamiltonian(const Graph& graph, double critParam)
{
    size_t N = graph.getVerticesCount();
    size_t firstPhaseEnd = N << 1;
    size_t secondPhaseEnd = firstPhaseEnd + N;

    std::vector<bool> visited(N);
    std::vector<size_t> recData;

    size_t time = 0;

    int currentGraphResult = backtrackingHamCycle(
        graph, 0, 0, visited, 0, time,
        recData, firstPhaseEnd, secondPhaseEnd
    );

    if (currentGraphResult > 0)
        return true;
    else if (currentGraphResult == 0)
        return false;
    else // currentGraphResult < 0
    {
        double average = 1.0 * std::accumulate(
            recData.begin(), recData.end(), 0LL
        ) / recData.size();
        double critLevel = N * (1 - critParam / log(N));
        return (average > critLevel);
    }
}

```

Променливата `critParam` на функцията `isHamiltonian` е числителят на умалителя в скобите на израза за равнището, около което се колебае траекторията на случайния процес. Както установихме по-рано, стойността `critParam = 1,52351` се намира в много добро съгласие с експерименталните данни за графи с 50 върха и плътности 7% и 17%. При провеждане на тестовете с тази стойност (`critParam = 1,52351`) алгоритъмът позна успешно 99,9684% от графите; останалите 0,0316% се оказаха все грешки от втория тип.

Последното обстоятелство е изненадващо, в смисъл че се очаква двата типа грешки да са разпределени поравно при критична плътност. Това, че грешките са все от втория тип, показва, че дясната страна на формулата

$$k_{\text{ср.}} \approx 50 \left( 1 - \frac{1,52351}{\ln 50} \right)$$

дава занижена оценка на лявата страна (типичното количество памет при графи с критична плътност). Дясната страна може да се завиши, като намалим стойността на `critParam` (числителя на умалителя). Това би трябвало да намали общия процент на грешките и да доведе до тяхното разделяне поравно от двата типа.

Действително, повторни тестове, с `critParam = 1,5`, показаха, че намаляването на числителя на `1,5` покачва успеваемостта на `99,9705 %`, съответно снижава процента на грешките на `0,0295 %`, но те все още са само от втория тип.

За да не търсим подходящата стойност на `critParam` с налущване, трябва да намерим източника на неточност. За формулата

$$k_{\text{ср.}} \approx 50 + \frac{1,52351}{\ln(1-p)}$$

проверихме с помощта на експерименталните данни, събрани по-рано, че дава добро приближение на типичното количество използвана памет. При малки  $p$  важи приближението  $\ln(1-p) \sim -p$ , следователно

$$k_{\text{ср.}} \approx 50 - \frac{1,52351}{p}.$$

По-нататък за  $p = p_{\text{кр.}}$  използвахме формулата на Поза и Коршунов:

$$p_{\text{кр.}} \approx \frac{\ln 50}{50}$$

и след заместване получихме

$$k_{\text{ср.}} \approx 50 \left( 1 - \frac{1,52351}{\ln 50} \right).$$

Но формулата на Поза и Коршунов, въпреки че е асимптотично точна, дава занижена оценка на критичната плътност на графа при  $n = 50$ : предвижда `8 %` вместо правилните `12 %`.

Тази неточност е равносилна на въвеждането на грешен множител  $8/12 = 2/3$  в знаменателя  $p$  на дробта. За компенсация би трябвало да добавим същия множител в числителя, тоест да променим числителя на  $(2/3) \cdot 1,52351 \approx 1$ . Тези разсъждения водят до предположението, че подходящата стойност на управляващия параметър е около 1.

Наистина, при  $\text{critParam} = 1$  алгоритъмът разпознава правилно 99,9937% от графите, а останалите 0,0063% са разпределени поравно между двата типа грешки, както би трябвало да бъде. Че тази стойност на управляващия параметър е по-точна от първоначалната, личи от това, че води до петкратно намаление на процента на грешните отговори.

По-нататъшното намаляване на стойността на  $\text{critParam}$  води до увеличаване на броя на грешките. Например при  $\text{critParam} = 0,8$  алгоритъмът разпознава правилно 99,9789% от входните данни, а останалите 0,0211% са само грешки от първия тип, което показва, че оптималната стойност на параметъра е вече подмината.

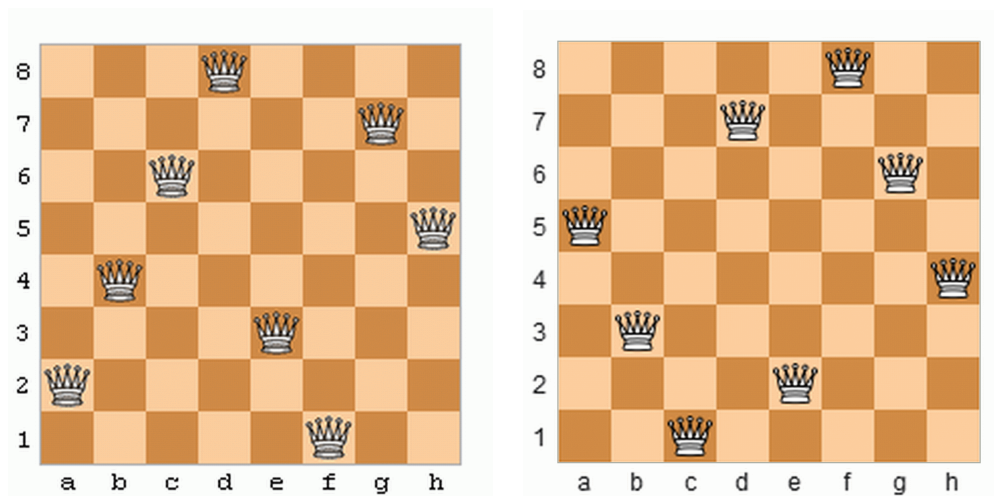
И така, в резултат от направените тестове можем да заключим, че оптимизираният алгоритъм съчетава бързодействие и надеждност. Нуждата от допълнителна настройка на управляващия параметър се дължи не на приближенията в нашия модел (той работи добре, в случай че е известна плътността  $p$ ), а на ограничената точност, с която формулата на Поза и Коршунов дава критичната плътност. Тяхната формула е асимптотично точна, но при не много големи  $n$  относителната грешка може да бъде значителна. Току-що видяхме, че например при  $n = 50$  се получава относителна грешка около  $1/3$ : наложи се да добавим поправъчен множител  $2/3$ . Той не е константа: тъй като формулата на Поза и Коршунов е асимптотично точна, то поправъчният множител клони към 1 при  $n \rightarrow \infty$ . Следователно неточността, породена от тази формула, изобщо не може да се поправи чрез добавяне на константен множител в израза за критичната плътност, а чрез добавяне на събираеми от по-нисък порядък. Откриването им се явява вътрешен проблем на теорията на хамилтоновите графи, ето защо излиза извън обхвата на нашето изследване. Целта ни тук е да създадем подходящ стохастичен модел на търсенето с връщане, приложим към различни алгоритмични задачи. Използването на модела в конкретна задача изисква, разбира се, конкретни познания за нея (например достатъчно точна формула за плътността), но липсата им е проблем на теорията на съответната задача, а не на нашия модел.

NP-пълната задача за разпознаване на хамилтонови графи беше разгледана по-горе като илюстрация на възможностите на модела. По-нататъшното ѝ изследване е извън целите на дипломната работа. Стохастичният модел на търсенето с връщане се прилага успешно и към други алгоритмични задачи, като обяснява закономерностите в използването на паметта и предлага възможност за сравняване на една задача с друга. Предстои да се уверим в това.

## 2 Задачата за цариците

В популярната математика е отдавна известна задачата за цариците: Върху шахматна дъска  $n \times n$  да се разположат  $n$  царици по такъв начин, че никои две от тях да не се бият.

За историята и резултатите по тази задача има обширна литература, включително едноименната статия от *Уикипедия* [7]. При  $n = 8$  задачата има 92 решения. Две от тях са показани тук.



Броя на решенията върху шахматна дъска  $n \times n$  означаваме с  $Q(n)$ . Разположенията, получени едно от друго чрез въртене или отражение, ще смятаме за различни. Например  $Q(8) = 92$ . Дъските  $2 \times 2$  и  $3 \times 3$  не притежават решения, тоест  $Q(2) = 0$  и  $Q(3) = 0$ .

*Енциклопедията на целочислените редици* [8] съдържа сведения за редицата  $(Q(n))_{n=1}^{\infty}$  под номер A000170. Точните стойности на  $Q(n)$  се знаят само за малки  $n$ . При  $n > 27$  не са известни нито стойности, нито точна или приблизителна формула за величината  $Q(n)$ . Непозната е дори асимптотиката на функцията; за нея има само предположения. Намерени са формули (например долни граници за броя на решенията) както за оригиналната задача, така и за нейни разновидности [9]. Някои от тези формули са доказани, а други — не. В една хипотеза на Беноа Кльоатър се твърди, че

$$Q(n) \sim \frac{n!}{c^n}$$

за някаква константа  $c \approx 2,54$ .

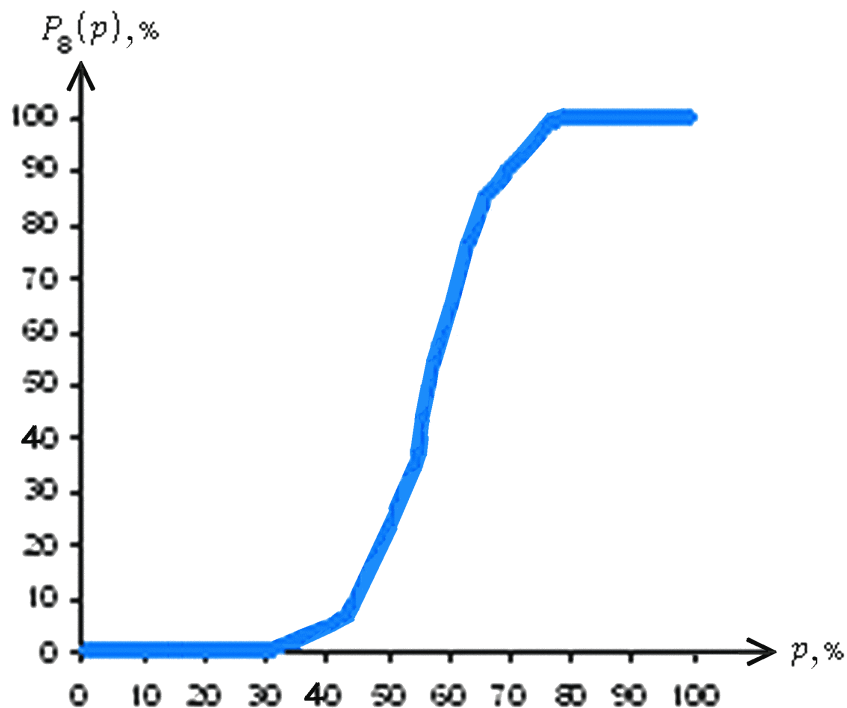
$n$	$Q(n)$	$(n! / Q(n))^{1/n}$
1	1	1,00
2	0	$\infty$
3	0	$\infty$
4	2	1,86
5	10	1,64
6	4	2,38
7	40	2,00
8	92	2,14
9	352	2,16
10	724	2,34
11	2680	2,40
12	14200	2,38
13	73712	2,39
14	365596	2,42
15	2279184	2,42
16	14772512	2,42
17	95815104	2,43
18	666090624	2,44
19	4968057848	2,45
20	39029188884	2,45
21	314666222712	2,46
22	2691008701644	2,47
23	24233937684440	2,47
24	227514171973736	2,47
25	2207893435808352	2,48
26	22317699616364044	2,48
27	234907967154122528	2,48

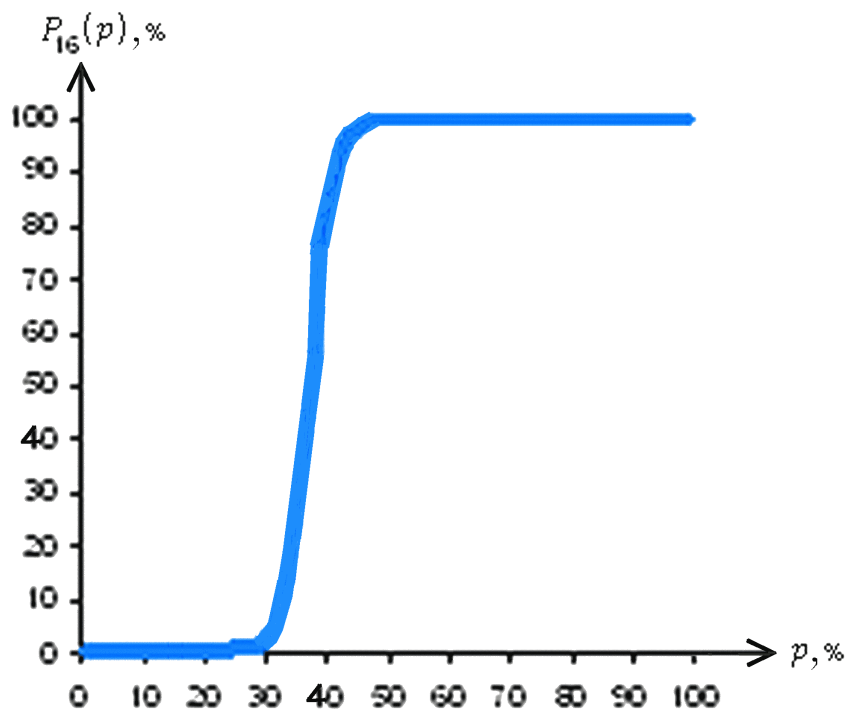


Приведената таблица съдържа всички известни стойности на  $Q(n)$ . Дължината на десетичния запис на  $Q(n)$  изглежда от порядък  $\Theta(n)$ , т.е.  $Q(n)$  расте експоненциално, а според хипотезата на Кльоатър — дори по-бързо, само че разликата не личи добре по десетичния запис: според хипотезата дължината му е  $\Theta(n \log n)$ , а не  $\Theta(n)$ , но разликата се забелязва трудно на око. Последният стълб на таблицата съдържа последователни приближения за числото  $c$  от хипотезата на Кльоатър. В нейна подкрепа редицата, изглежда, клони към число около 2,5.

Тъй като задачата има решение за всяко  $n$ , различно от 2 и 3, уместно е да въведем параметър  $p$  — плътност на шахматната дъска. Ще смятаме, че някои от полетата на шахматната дъска са забранени: върху тях не можем да слагаме царици. Ще генерираме входните данни по случаен начин, като всяко поле ще бъде разрешено с вероятност  $p$  независимо от другите полета на дъската. Допустими са  $p \in [0; 1]$ .

Означаваме с  $P_n(p)$  вероятността за съществуване на разположение на  $n$  царици, без да се бият, върху шахматна дъска  $n \times n$  с плътност  $p$ . Очевидно  $P_n(0) = 0$ ,  $P_n(1) = 1$  и  $P_n(p)$  е растяща функция на  $p$ .





Експериментите показват, че функцията  $P_n(p)$  приема предимно стойности, близки до 0 или 1. Изключение прави един критичен участък, в който  $P_n(p)$  бързо нараства от 0 до 1. Извън критичния участък резултатът от търсенето е известен предварително с голяма вероятност:

— Ако плътността  $p$  се намира надясно от критичния участък, то почти сигурно съществува разположение на  $n$  царици.

— Ако плътността  $p$  се намира наляво от критичния участък, то почти сигурно не съществува разположение на  $n$  царици.

Така че задачата представлява интерес само за шахматни дъски с плътност от критичния участък на  $P_n(p)$ . Експериментите показват, че с нарастването на  $n$  критичният участък се свива и се мести наляво. Затова е важно да познаваме асимптотичното му поведение при  $n \rightarrow \infty$ . За целта въвеждаме понятието критична плътност:

$$p_{\text{кр.}} = P_n^{-1}(0,5).$$

Това е плътността, при която отговорът на задачата е най-несигурен. Критичната плътност е функция на  $n$ . От графиките на  $P_n(p)$  се вижда, че  $p_{\text{кр.}}$  е между 0,57 и 0,59 при  $n = 8$ ;  $p_{\text{кр.}} \approx 0,37$  при  $n = 16$ .

При задачата за цариците липсва аналог на теоремата на Поза, (асимптотична формула за критичната плътност на шахматната дъска). От хипотезата на Беноа Кльоатър чрез не съвсем строги разсъждения може да се изведе приближение за критичната плътност.

По определение има  $Q(n)$  разположения на  $n$  царици върху дъска, чиито полета са само разрешени. Ако всяко поле независимо от другите остане разрешено с вероятност  $p$  или се забрани с вероятност  $1 - p$ , то разположенията ще намалеят и дори може да изчезнат напълно. Вероятността да се запази едно конкретно разположение е равна на вероятността да останат разрешени всичките му полета, а тя е  $p^n$  според правилото за умножение на вероятности (можем да го приложим, защото изборът за всяко поле — дали е разрешено, или забранено — се прави независимо от останалите полета).

Вероятността да отпадне едно конкретно разположение е  $1 - p^n$ .

Следва нестрого разсъждение. Пак умножаваме вероятностите и стигаме до извода, че вероятността да отпаднат всички разположения е приблизително  $(1 - p^n)^{Q(n)}$ . Това разсъждение е нестрого, тъй като разположенията с общи полета са зависими и изискват да се работи с условни вероятности, а те зависят от броя на общите полета. Получават се сложни изрази, трудни за обработка. Както казахме, не е доказана даже асимптотична формула за задачата за цариците, затова ще използваме полученото приближение, като го сверим с експерименталните резултати. Като слаба теоретична обосновка можем да изтъкнем следните съображения: повечето двойки и по-големи множества от разположения имат малко общи полета, откъдето следва, че такива разположения са слабо зависими, така че изводът важи за тях поне приблизително. Обратно, разположения с много общи полета са силно зависими, но се срещат рядко, така че имат малък принос към крайния резултат. Това показва, че изведеното приближение може би ще се окаже приемливо поне в асимптотичен смисъл.

Изразът  $(1 - p^n)^{Q(n)}$  представлява приближение за вероятността да отпаднат всички разположения, тоест задачата да няма решение. Следователно

$$P_n(p) \approx 1 - (1 - p^n)^{Q(n)}$$

е вероятността да съществува поне едно разположение на  $n$  царици върху шахматна дъска  $n \times n$  с плътност  $p$ .

В полученото приближение заместяваме  $p = p_{\text{кр.}}$  :

$$P_n(p_{\text{кр.}}) \approx 1 - \left(1 - (p_{\text{кр.}})^n\right)^{Q(n)}.$$

По определение  $P_n(p_{\text{кр.}}) = 0,5$ . Следователно

$$\left(1 - (p_{\text{кр.}})^n\right)^{Q(n)} \approx \frac{1}{2}.$$

Решаваме това уравнение относно  $p_{\text{кр.}}$  :

$$p_{\text{кр.}} \approx \left(1 - 2^{-1/Q(n)}\right)^{1/n}.$$

От хипотезата на Беноа Кльоатър заместяваме

$$Q(n) \sim \frac{n!}{c^n}$$

и намираме критичната плътност на задачата за цариците:

$$p_{\text{кр.}} \approx \left(1 - 2^{-(c^n)/n!}\right)^{1/n}.$$

Този израз е явна формула за критичната плътност като функция на  $n$ , но все пак не е много удобен за работа. Ако искаме например да намерим границата на критичната плътност при  $n \rightarrow \infty$ , то от тази формула получаваме неопределеността  $[0^0]$ . Затова ще преработим формулата, като се върнем с няколко стъпки:

$$\left(p_{\text{кр.}}\right)^n \approx 1 - 2^{-(c^n)/n!} \rightarrow 0 \text{ при } n \rightarrow \infty.$$

Следователно

$$\left(1 - (p_{\text{кр.}})^n\right)^{n!/c^n} \approx \frac{1}{2}.$$

Преработваме лявата страна:

$$\left\{ \left(1 - (p_{\text{кр.}})^n\right)^{1/(p_{\text{кр.}})^n} \right\}^{n! (p_{\text{кр.}})^n / c^n} \approx \frac{1}{2}.$$

Тъй като  $(p_{\text{кр.}})^n \rightarrow 0$  при  $n \rightarrow \infty$ , то изразът във фигурните скоби клони към  $1/e$ . Следователно при големи  $n$  важи приближението

$$\left(\frac{1}{e}\right)^{n! (p_{\text{кр.}})^n / c^n} \approx \frac{1}{2}.$$

Оттук

$$\frac{n! (p_{\text{кр.}})^n}{c^n} \approx \ln 2.$$

Коренуваме:

$$\frac{\sqrt[n]{n!} p_{\text{кр.}}}{c} \approx \sqrt[n]{\ln 2}.$$

Понеже  $\sqrt[n]{n!} \sim n/e$  (от формулата на Стирлинг) и  $\sqrt[n]{\ln 2} \rightarrow 1$  при  $n \rightarrow \infty$ , то получената формула се опростява така:

$$\frac{n p_{\text{кр.}}}{e c} \approx 1.$$

Това уравнение лесно се решава относно критичната плътност:

$$p_{\text{кр.}} \approx \frac{e c}{n}.$$

Тук  $e \approx 2,72$  е основата на натуралните логаритми, а пък  $c \approx 2,54$  е константата от хипотезата на Кльоатър. Затова  $e c \approx 6,9 \approx 7$  и

$$p_{\text{кр.}} \approx \frac{7}{n}.$$

При  $n = 8$  приближението е изключително неточно:  $p_{\text{кр.}} \approx 0,875$  (истинската стойност е между 0,57 и 0,59). Тази неточност е естествена: все пак формулата е изведена при предположението, че  $n \rightarrow \infty$ . Затова тя би могла да бъде вярна единствено в асимптотичен смисъл.

При  $n = 16$  приближението дава по-точен резултат:  $p_{\text{кр.}} \approx 0,44$  (истинската стойност е около 0,37).

Вижда се, че с нарастването на  $n$  формулата става все по-точна. Простотата ѝ е съществено предимство, затова ще я използваме за асимптотична оценка на поведението на алгоритъма.

Когато плътността на шахматната дъска е около критичната, използваме търсене с връщане. Разполагаме цариците една по една, например отляво надясно, тоест стълб по стълб. Нека сме поставили  $\xi$  царици в първите  $\xi$  стълба така, че да не се бият. В стълб №  $\xi + 1$  търсим място за следващата царица. Обхождаме стълба, търсейки поле, което не е под удар. Ако намерим такова поле, слагаме царица върху него и минаваме към следващия стълб. В противен случай махаме царицата, сложена последна (тоест №  $\xi$ ) и търсим друго подходящо за нея поле в стълб №  $\xi$ . Ако не намерим такова, се връщаме още един стълб наляво: премахваме царицата от стълб №  $\xi - 1$  и търсим за нея друго поле в същия стълб. В крайна сметка намираме разположение на цариците или изчерпваме всички възможности в първия стълб, което е признак, че задачата няма решение при тези входни данни.

Търсенето на място за поредната царица чрез обхождане на стълб ще смятаме за една стъпка от алгоритъма. С буквата  $\xi$  ще означаваме текущия брой на цариците, поставени върху шахматната дъска  $n \times n$ . Разбира се, броят им зависи от стъпката, затова ще пишем индекс  $t$ , тоест  $\xi_t$  е броят на цариците върху дъската след стъпка №  $t \geq 1$ . Въвеждаме и величината  $\xi_0$  — начален брой на цариците върху дъската. Очевидно  $\xi_0 = 0$ .

Редицата  $(\xi_t)_{t \geq 0}$  може да бъде тълкувана като случаен процес дори ако алгоритъмът е детерминиран. Причината е, че входните данни (разрешените и забранените полета на дъската) могат да се разглеждат като случайни. За да бъде алгоритъмът по-близък до този модел, можем да въведем елемент на случайност в самия алгоритъм. Например търсенето на място за поредната царица може да става чрез обхождане на полетата на съответния стълб в случаен ред.

Алгоритъмът може да се реализира итеративно или рекурсивно. При рекурсивна реализация  $\xi_t$  е текущата дълбочина на рекурсията, а тя е правопрпорционална на количеството допълнителна памет, използвано от алгоритъма. В този смисъл  $\xi_t$  може да се тълкува и като количество памет.

Ясно е, че  $0 \leq \xi_t \leq n$  и  $\xi_{t+1} = \xi_t \pm 1$  за всяко  $t$ . Полагаме

$$p_{n,k} \stackrel{\text{def}}{=} \mathbf{P}\{\xi_{t+1} = \xi_t - 1 \mid \xi_t = k\} = \mathbf{P}\{\xi_{t+1} = k - 1 \mid \xi_t = k\}.$$

Поставяме си за цел да намерим формула за тази вероятност.

Нека  $\zeta$  е броят на полетата в стълб №  $k + 1$ , които не се намират под удара на никоя от цариците в първите  $k$  стълба. Числото  $\zeta$  зависи от разположението на споменатите  $k$  царици, т.е.  $\zeta$  е случайна величина. Вероятността да са забранени всичките  $\zeta$  полета е равна на  $(1 - p)^\zeta$  според правилото за умножение на вероятности (защото всяко поле е разрешено или забранено независимо от останалите полета), а пък  $p_{n;k}$  е усреднената стойност (математическото очакване) на горния израз:

$$p_{n;k} = \mathbf{E}\left((1 - p)^\zeta\right) \approx \left(1 - \frac{p \mathbf{E}(\zeta)}{n}\right)^n.$$

Второто равенство е вярно, защото средният израз съвпада по същество с характеристикната функция на биномното разпределение с параметри  $n$  (броя на опитите, т.е. броя на полетата в стълб №  $k + 1$  на дъската) и  $\frac{\mathbf{E}(\zeta)}{n}$  (вероятност за успех, т.е. оцеляване на поле). Величината  $\zeta$  има приблизително биномно разпределение с посочените параметри. (приблизително, защото вероятността за оцеляване не е една и съща за всички полета от последния стълб).

За да пресметнем очакването  $\mathbf{E}(\zeta)$ , трябва да вземем предвид, че върху всяко поле в стълб №  $k + 1$  може да попаднат три вида удари от цариците, разположени в първите  $k$  стълба:

- водоравни (надясно);
- по диагонал надолу и надясно;
- по диагонал нагоре и надясно.

Ще оценим приблизително каква част от полетата в стълб №  $k + 1$  оцелява след всеки вид удари. Без ограничение можем да приемем, че видовете удари се нанасят в реда, в който са изброени по-горе. От всеки вид се нанасят точно  $k$  удара (обаче е възможно някои удари да не попаднат върху поле от стълб №  $k + 1$ ).

Водоравните удари засягат точно  $k$  квадратчета от стълб №  $k + 1$  и след тях оцеляват точно  $n - k = n \left(1 - \frac{k}{n}\right)$  полета от този стълб.

Изразът в скобите показва частта на оцелелите квадратчета, тоест вероятността за оцеляване на случайно избрано поле от стълб №  $k + 1$ . Вероятностният подход е приложим и за другите видове удари.

1	■	■	■	■	■	■
2	■	■	■	■	■	■
⋮	■	■	■	■	■	■
$k$	■	■	■	■	■	■
$k+1$	■	■	■	■	■	■
$k+2$	■	■	■	■	■	■
⋮	■	■	■	■	■	■
⋮	■	■	■	■	■	■
⋮	■	■	■	■	■	■
⋮	■	■	■	■	■	■
⋮	■	■	■	■	■	■
$n$	■	■	■	■	■	■
	1	2	⋯	$k$	$k+1$	

Нека  $A = n - k = n \left(1 - \frac{k}{n}\right)$  е броят на полетата в стълб №  $k + 1$ , оцелели след водоравните удари. Идва ред на втория вид удари — по диагонал надолу и надясно. За да ги анализираме по-лесно, е удобно да разделим цариците на три вида: жълти, сини и розови (според това, какво място заемат в правоъгълника, съставен от първите  $k$  стълба и всичките  $n$  реда на шахматната дъска). Броят на цариците от всеки вид е случайна величина, чието математическо очакване е приблизително пропорционално на лицето на съответната област от правоъгълника. Пропорцията би била точна, ако цариците заемаха местата си случайно и независимо една от друга. В действителност те избягват взаимни удари, което води до тяхното раздалечаване, тоест по-равномерно разпръскване из правоъгълника  $k \times n$ . Това не влияе много на средните стойности, а най-вече намалява дисперсията.

Първите  $k$  стълба на дъската съдържат общо  $kn$  полета, от които точно  $k(n - k)$  са сини,  $1 + 2 + 3 + \dots + (k - 1) = \frac{(k - 1)k}{2}$  са розови и  $1 + 2 + 3 + \dots + k = \frac{k(k + 1)}{2}$  са жълти. След деление на общия брой от абсолютните бройки се получават относителните части на цветовете:  $1 - \frac{k}{n}$  сини,  $\frac{k - 1}{2n}$  розови и  $\frac{k + 1}{2n}$  жълти полета.



Както казахме, ще приемем, че тези числа са относителни части и на трите цвята царици. Първите  $k$  стълба съдържат общо  $k$  царици и  $\frac{k(n-k)}{n}$  от тях са сини,  $\frac{k(k-1)}{2n}$  са розови и  $\frac{k(k+1)}{2n}$  са жълти, като това са приблизителни, средни бройки.

Всяка от тези  $k$  царици отправя по един удар надолу и надясно (общо  $k$  удара). Всички тези удари попадат в различни квадратчета от стълб №  $k+1$  или неговото продължение, тъй като цариците, поставени на дъската, не се атакуват взаимно, затова никои две от тях не лежат на един и същи диагонал.

Ударите на жълтите царици уцелват не самия стълб №  $k+1$ , а продължението му, затова не намаляват броя на оцелелите полета.

Ударите на сините царици — приблизително  $\frac{k(n-k)}{n}$  на брой — уцелват толкова от всичките  $n-k$  светлосиви полета на стълб №  $k+1$ . Следователно вероятността светлосиво поле да оцелее след тези удари е равна на  $1 - \frac{k}{n}$ .

Светлосивите полета са  $\frac{n-k}{n} = 1 - \frac{k}{n}$  част от всички сиви полета, затова около  $A \left(1 - \frac{k}{n}\right)$  от тях са оцелели след водоравните удари.

След ударите от втория вид оцеляват  $A \left(1 - \frac{k}{n}\right)^2$  светлосиви полета средно.

Ударите на розовите царици — приблизително  $\frac{k(k-1)}{2n}$  на брой — уцелват толкова от всичките  $k$  тъмносиви полета на стълб №  $k+1$ . Следователно вероятността тъмносиво поле да оцелее след тези удари е равна на  $1 - \frac{k-1}{2n}$ .

Тъмносивите полета са  $\frac{k}{n}$  част от всички сиви, поради което около  $\frac{k}{n}A$  тъмносиви квадратчета са оцелели след водоравните удари.

След ударите от втория вид оцеляват приблизително  $A \left(1 - \frac{k-1}{2n}\right) \frac{k}{n}$  тъмносиви полета.

Да означим с  $B$  средния брой оцелели полета от стълб №  $k + 1$  след ударите от втория вид. Току-що направените разсъждения показват, че

$$B \approx A \left(1 - \frac{k}{n}\right)^2 + A \left(1 - \frac{k-1}{2n}\right) \frac{k}{n}.$$

След преобразуване този израз добива вида:

$$B \approx A \left(1 - \frac{k}{n} + \frac{k(k+1)}{2n^2}\right).$$

Изразът в скобите е частта на полетата, оцелели след втория вид удари.

Този резултат е изведен при предположението за независимост на различните видове удари. Ако водоравно ударени са например 80 %, а под удар от втория вид са например 70 % от полетата в стълб №  $k + 1$ , то полетата, уцелени два пъти, са около 70 % от 80 %, тоест 56 %, според предположението за независимост. Само че то не е вярно: разликата в посоките на ударите води до намаляване на сечението им.

Възниква въпросът как да измерим това разпръскване на ударите. Забелязваме, че водоравните удари разполагат с  $k$  от общо  $n$  линии — редовете на правоъгълника; а пък ударите по диагонал надолу и надясно се движат по  $k$  от общо  $n + k$  линии — диагоналите надолу и надясно. Допълнителните  $k$  линии пресичат продължението на стълб №  $k + 1$  в  $k$  полета; те не са нанесени на чертежа по-горе, защото са не от стълба, а от неговото продължение надолу — там, където попадат ударите на жълтите царици по диагонал надясно и надолу.

Разбира се, тези удари не намаляват броя на оцелелите полета в стълб №  $k + 1$ , защото уцелват не самия стълб, а продължението му. Този факт беше отчетен в сметките по-горе. Но сега се нуждаем от оценка за разпръскването на ударите. То действа и в рамките на стълб №  $k + 1$ , а допълнителните  $k$  полета от продължението на стълба са само мярка за големината на разпръскването: въпреки че продължението на стълба не съдържа квадратчета, които могат да бъдат уцелени, все пак броят на полетата, които лежат в продължението на стълба с  $k$  клетки надолу и биха били уцелени по диагонал надолу и надясно, се явява оценка за разпръскването и трябва да се прибави към броя на уцелените полета (всъщност лежащи в самия стълб №  $k + 1$ , а не в продължението му).

Ударите на жълтите царици по диагонал надолу и надясно уцелват  $\frac{k(k+1)}{2n}$  от общо  $k$  полета в продължението на стълб №  $k+1$  надолу, т.е. вероятността едно конкретно поле от тези да попадне под такъв удар е равна на  $\frac{k+1}{2n}$ .

От друга страна, не всичките  $k$  полета от продължението са оцелели след водоравните удари. Върху продължението на последния стълб не попадат водоравни удари, но продължението се разглежда само с цел оценка на разпръскването на ударите. Допълнително уцелените полета, чийто брой търсим, лежат в самия стълб №  $k+1$ , не в продължението му. Затова трябва да смятаме, че след водоравните удари е оцеляла част  $\frac{A}{n}$ , тоест  $\frac{kA}{n}$  полета от разглежданото продължение. Сред тези  $\frac{kA}{n}$  полета приблизително  $\frac{k+1}{2n} \cdot \frac{kA}{n} = \frac{k(k+1)A}{2n^2}$  попадат под диагонален удар надолу и надясно.

Тази бройка трябва да се добави към броя на ударените полета, следователно трябва да се извади от броя на оцелелите:

$$B \approx A \left( 1 - \frac{k}{n} + \frac{k(k+1)}{2n^2} \right) - \frac{k(k+1)A}{2n^2},$$

откъдето следва, че

$$B \approx A \left( 1 - \frac{k}{n} \right).$$

Това е средният брой на полетата в последния стълб, които оцеляват след първите два вида удари.

Аналогично

$$\mathbf{E}(\zeta) \approx B \left( 1 - \frac{k}{n} \right)$$

е средният брой полета в стълб №  $k+1$ , оцелели сред третия вид удари, тоест полетата, които не са попаднали под никакъв удар. Заместваме първо  $B$ , после  $A$  и установяваме, че

$$\mathbf{E}(\zeta) \approx n \left( 1 - \frac{k}{n} \right)^3.$$

Заместваме този резултат във формулата

$$p_{n;k} \approx \left(1 - \frac{p \mathbf{E}(\zeta)}{n}\right)^n$$

и намираме

$$p_{n;k} \approx \left(1 - p \left(1 - \frac{k}{n}\right)^3\right)^n, \quad 0 \leq k \leq n.$$

Случаят  $p = 0$  не е интересен: тогава шахматната дъска почти сигурно се състои само от забранени полета и не съществува разположение на  $n$  царици. Затова нека  $p \neq 0$ , тоест  $p \in (0; 1]$ . В такъв случай  $p_{n;k}$  е строго растяща функция на  $k$ . Ако допустимите стойности на  $k$  (числата  $0, 1, 2, \dots, n$ ) разположим върху абсцисната ос и тълкуваме  $p_{n;k}$  като физична сила, то въпросната сила тласка случайния процес  $(\xi_t)_{t \geq 0}$  надясно при малки  $k$  и наляво при големи  $k$ . Ето защо съществува точка на устойчиво равновесие, около която  $\xi_t$  извършва движение, подобно на случайно лутане; това е точката  $k_{0,5}$ , за която  $p_{n;k_{0,5}} = 0,5$ . Като решим относно  $k_{0,5}$  уравнението

$$\left(1 - p \left(1 - \frac{k_{0,5}}{n}\right)^3\right)^n \approx \frac{1}{2},$$

намираме

$$k_{0,5} \approx n \left(1 - \sqrt[3]{\frac{1 - 2^{-1/n}}{p}}\right).$$

Търсенето с връщане работи дълго само при входни данни, за които резултатът не е достатъчно сигурен, т.е. за плътности около критичната:

$$p \approx p_{\text{кр.}} \approx \frac{7}{n}.$$

Тогава

$$k_{0,5} \approx n \left(1 - \sqrt[3]{\frac{1 - 2^{-1/n}}{7/n}}\right).$$

При големи входни данни, тоест при  $n \rightarrow \infty$ , подкоренната величина клони към  $\frac{\ln 2}{7}$  и

$$k_{0,5} \approx n \left( 1 - \sqrt[3]{\frac{\ln 2}{7}} \right).$$

Изразът в скобите има стойност около 0,537. Но като вземем предвид характера на другите направени приближения, става ясно, че можем да разчитаме само на първата значеща цифра в резултата, т.е. добре е да приемем приближението 0,5. Следователно

$$k_{0,5} \approx \frac{n}{2}.$$

Това е равновесната точка, която привлича траекториите на процеса. Тя може да не съвпада с математическото очакване на стойностите от водоравната част на траекториите.

Като решим относно  $k_\gamma$  уравнението  $p_{n;k_\gamma} = \gamma$ , тоест

$$\left( 1 - p \left( 1 - \frac{k_\gamma}{n} \right)^3 \right)^n \approx \gamma,$$

намираме

$$k_\gamma \approx n \left( 1 - \sqrt[3]{\frac{1 - \gamma^{1/n}}{p}} \right).$$

Разглеждаме плътности около критичната:

$$p \approx p_{\text{кр.}} \approx \frac{7}{n}.$$

Тогава

$$k_\gamma \approx n \left( 1 - \sqrt[3]{\frac{1 - \gamma^{1/n}}{7/n}} \right).$$

При големи входни данни, тоест при  $n \rightarrow \infty$ , подкоренната величина клони към  $-\frac{\ln \gamma}{7}$  и

$$k_\gamma \approx n \left( 1 + \sqrt[3]{\frac{\ln \gamma}{7}} \right).$$

В околност на точката  $k_{0,5}$  случайният процес е симетрично лутане. От съображения за симетрия на връщащата сила въпросната околност трябва да бъде избрана от вида  $[k_{1-\gamma}; k_\gamma]$  за подходящо  $\gamma \in (0; 1)$ . Повтаряйки без промяна разсъжденията за хамилтонов цикъл (представяме случайното лутане чрез верига на Марков и изследваме нейното гранично разпределение), установяваме, че математическото очакване на стойностите от водоравната част на траекториите е точно средата на интервала

$$k_{\text{ср.}} = \frac{1}{2}k_\gamma + \frac{1}{2}k_{1-\gamma}.$$

Следователно

$$k_{\text{ср.}} \approx n \left( 1 - \frac{1}{2} \sqrt[3]{\frac{1 - \gamma^{1/n}}{p}} - \frac{1}{2} \sqrt[3]{\frac{1 - (1 - \gamma)^{1/n}}{p}} \right).$$

За плътности около критичната:

$$k_{\text{ср.}} \approx n \left( 1 - \frac{1}{2} \sqrt[3]{\frac{1 - \gamma^{1/n}}{7/n}} - \frac{1}{2} \sqrt[3]{\frac{1 - (1 - \gamma)^{1/n}}{7/n}} \right).$$

За голям брой върхове и плътности около критичната:

$$k_{\text{ср.}} \approx n \left( 1 + \frac{1}{2} \sqrt[3]{\frac{\ln \gamma}{7}} + \frac{1}{2} \sqrt[3]{\frac{\ln(1 - \gamma)}{7}} \right).$$

Параметърът  $\gamma$  може да се намери от уравнението

$$\sum_{k=k_{1-\gamma}+1}^{k_\gamma} \log_2 p_{n;k} = -n.$$

То се доказва по същия начин, както при търсене на хамилтонов цикъл. Тоест уравнението и доказателството му имат висока степен на общност. Разликата между двете задачи е само във вероятностите за преход. В задачата за цариците

$$p_{n;k} \approx \left( 1 - p \left( 1 - \frac{k}{n} \right)^3 \right)^n.$$

След заместване в уравнението то приема формата

$$\sum_{k=k_{1-\gamma}+1}^{k_\gamma} n \ln \left( 1 - p \left( 1 - \frac{k}{n} \right)^3 \right) \approx -n \ln 2.$$

Опростяваме лявата страна на уравнението и съкращаваме на  $-n$ :

$$\sum_{k=k_{1-\gamma}+1}^{k_\gamma} p \left( 1 - \frac{k}{n} \right)^3 \approx \ln 2.$$

Умножаваме по  $n^3$  и делим на  $p$ :

$$\sum_{k=k_{1-\gamma}+1}^{k_\gamma} (n - k)^3 \approx \frac{n^3}{p} \ln 2,$$

тоест

$$\sum_{k=n-k_\gamma}^{n-k_{1-\gamma}-1} k^3 \approx \frac{n^3}{p} \ln 2.$$

Асимптотиката не се променя, ако интегрираме, вместо да сумираме, и ако пропуснем умалителя 1:

$$\frac{\left( n - k_{1-\gamma} \right)^4 - \left( n - k_\gamma \right)^4}{4} \approx \frac{n^3}{p} \ln 2.$$

Полученото уравнение делим на  $n^4$  и умножаваме по 4:

$$\left( 1 - \frac{k_{1-\gamma}}{n} \right)^4 - \left( 1 - \frac{k_\gamma}{n} \right)^4 \approx \frac{4 \ln 2}{np}.$$

Заместваем  $k_\gamma$  и  $k_{1-\gamma}$  в уравнението:

$$\left( \frac{1 - (1 - \gamma)^{1/n}}{p} \right)^{4/3} - \left( \frac{1 - \gamma^{1/n}}{p} \right)^{4/3} \approx \frac{4 \ln 2}{np}.$$

Това уравнение определя  $\gamma$  като неявна функция на  $n$  и  $p$ .

За плътности около критичната, тоест за  $p \approx p_{\text{кр.}} \approx \frac{7}{n}$ , е възможно още по-голямо опростяване:

$$\left(\frac{1 - (1 - \gamma)^{1/n}}{7/n}\right)^{4/3} - \left(\frac{1 - \gamma^{1/n}}{7/n}\right)^{4/3} \approx \frac{4 \ln 2}{7}.$$

Умаляемото и умалителят в лявата страна имат граници при  $n \rightarrow \infty$ ; тези граници се пресмятат по правилото на Лопитал. Резултатът е, че при големи  $n$  важи приближението

$$\left(\frac{\ln(1 - \gamma)}{7}\right)^{4/3} - \left(\frac{\ln \gamma}{7}\right)^{4/3} \approx \frac{4 \ln 2}{7}.$$

Опростяваме полученото уравнение:

$$\ln^{4/3}(1 - \gamma) - \ln^{4/3} \gamma \approx 4 \sqrt[3]{7} \ln 2.$$

Лявата страна е строго растяща функция, поради което уравнението има единствено решение, което може да се намери числено:

$$\gamma \approx 0,96979.$$

За тази стойност на  $\gamma$  пресмятаме типичното количество памет:

$$k_{\text{ср.}} \approx n \left( 1 + \frac{1}{2} \sqrt[3]{\frac{\ln 0,96979}{7}} + \frac{1}{2} \sqrt[3]{\frac{\ln(1 - 0,96979)}{7}} \right)$$

Изразът в големите скоби притежава стойност  $0,52 \approx 0,5$ . Ето защо

$$k_{\text{ср.}} \approx k_{0,5} \approx \frac{n}{2}.$$

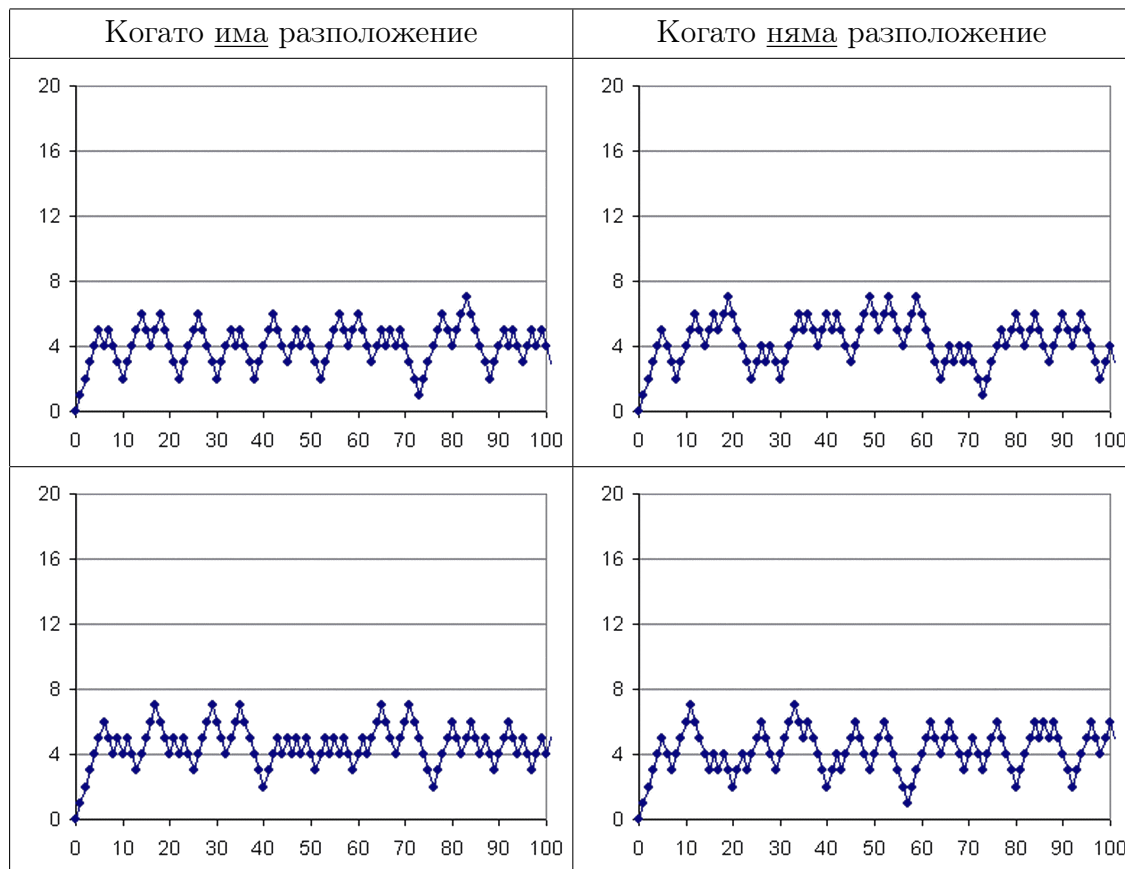
Около тази стойност се колебаят траекториите на случайния процес.

Изразът в скобите зависи от  $\gamma$  слабо, когато  $\gamma$  не е твърде близо до 1. Затова  $k_{\text{ср.}}$  и  $k_{0,5}$  практически съвпадат.

Намерената формула за средната стойност на процеса има подкрепа и от експериментите. Алгоритъмът търсене с връщане беше изпълнен върху шахматни дъски с различни размери. Беше събрана информация за траекториите на случайния процес при разнообразни входни данни. Нека разгледаме обобщените резултати от проведените наблюдения върху работата на алгоритъма.



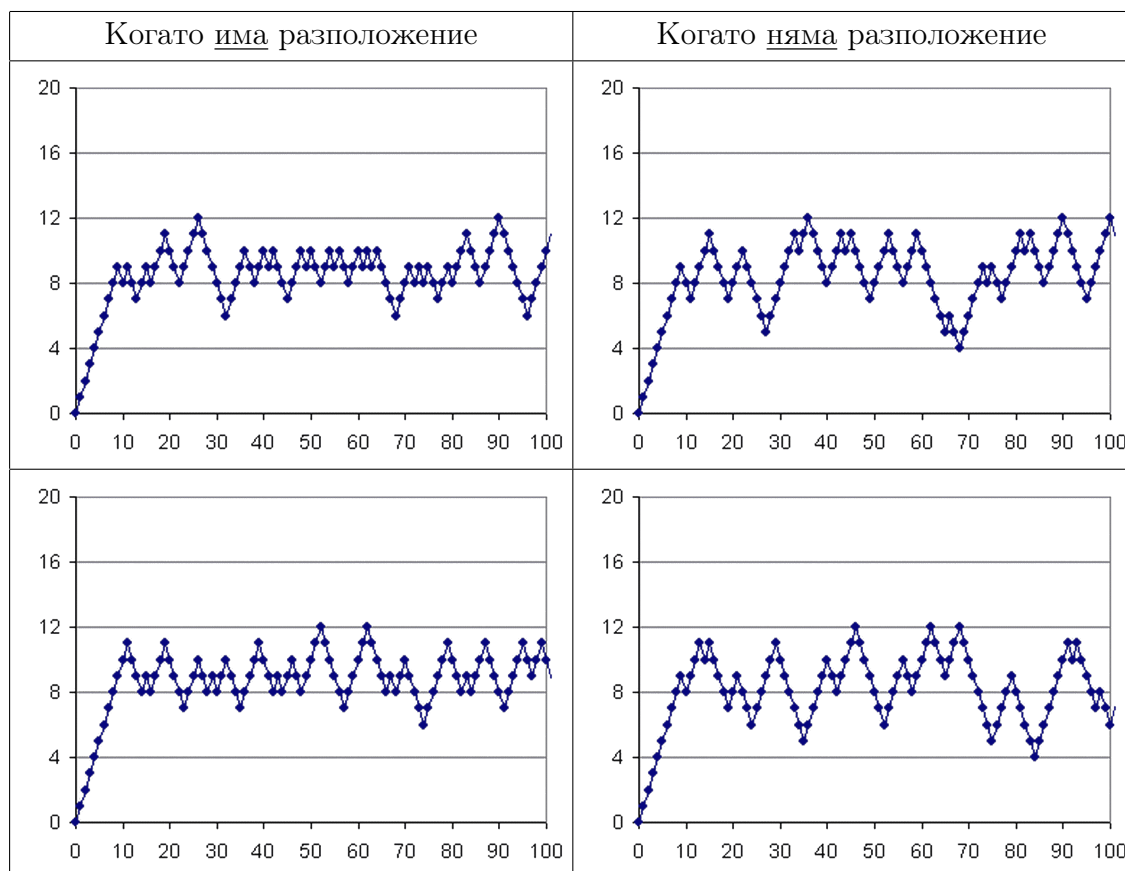
Показани са примерни траектории на случайния процес  $(\xi_t)_{t \geq 0}$  при шахматни дъски  $8 \times 8$  с плътности, равни на критичната (59%), когато може и когато не може да се разположат 8 царици, без да се бият.



Не се виждат големи разлики между траекториите от двата вида. Това личи и от проверката на съответните статистически хипотези. При извадка от данни, когато съществува разположение на цариците, с обем  $n_x = 24$ , средноаритметично  $\bar{x} = 4,73$  и отклонение  $s_x = 1,40$  и извадка от данни, когато не съществува разположение на цариците, с обем  $n_y = 35$ , средноаритметично  $\bar{y} = 4,47$  и отклонение  $s_y = 1,44$  наблюдаваната стойност на F-статистиката е 1,06, което съответства на огромна р-стойност: 0,9, т.е. няма значима разлика в отклоненията; наблюдаваната стойност на T-статистиката е 0,69, отново с голяма р-стойност: 0,5, т.е. няма значима разлика в средните стойности.

Забележка: Сметките се отнасят за двустранни алтернативи.

Същото се наблюдава и при входни данни с по-голяма дължина — при шахматни дъски  $16 \times 16$  с плътности, равни на критичната (37%). Показани са примерни траектории на случайния процес в двата случая — когато може и когато не може да се разположат 16 царици, без да се бият.



Отново няма големи разлики между траекториите от двата вида. При извадка от данни, когато съществува разположение на цариците, с обем  $n_x = 30$ , средноаритметично  $\bar{x} = 8,78$  и отклонение  $s_x = 1,67$  и извадка от данни, когато не съществува разположение на цариците, с обем  $n_y = 37$ , средноаритметично  $\bar{y} = 8,12$  и отклонение  $s_y = 1,82$  наблюдаваната стойност на F-статистиката е 1,19, което съответства на твърде голяма р-стойност: 0,64, т.е. няма разлика в отклоненията; наблюдаваната стойност на T-статистиката е 1,53 с р-стойност: 0,13, която, въпреки че не е много голяма, надхвърля най-често използваните нива на доверие (вкл. 0,1), т.е. можем да приемем равенство на средните.

Стандартното отклонение е едно и също в двата случая:

$$\sigma \approx \frac{1}{2}\sqrt{n}.$$

Макар че тази формула бе изведена за търсенето на хамилтонов цикъл, тя важи и в задачата за цариците, както показва числената проверка: при  $n = 8$  се получава  $\sigma \approx 1,4$ , а при  $n = 16$  се получава  $\sigma \approx 2,0$ . Приближението е с излишък (при големи  $n$ ), но в задачата за цариците то е по-точно, отколкото в задачата за хамилтоновите цикли.

Обяснението за валидността на формулата е, че разсъжденията, с които беше изведена, имат голяма общност. Случайният процес се колебае около някаква стойност между 0 и  $n$  със стъпки  $\pm 1$ , като двата вида стъпки са равновероятни, тоест колебанията имат характер на симетрично случайно лутане независимо от задачата.

Точката  $k$  на устойчиво равновесие е единственото решение на уравнението  $p_{n;k} = 0,5$ . Формулата за отклонението е по-точна, когато процесът е по-близък до симетрично случайно лутане, т.е. когато  $p_{n;k} \approx 0,5$  в по-широк интервал около точката на устойчиво равновесие. Това е изпълнено, когато  $p_{n;k}$  се мени по-бавно с изменянето на  $k$ , тоест когато частното

$$\frac{p_{n;k}}{p_{n;k+1}}$$

е по-близо до единицата.

Когато използваме алгоритмичната схема търсене с връщане в задачата за разпознаване на хамилтонови от нехамилтонови графи, вероятността за неуспешна стъпка е равна на

$$p_{n;k} = (1 - p)^{n-k},$$

следователно частното, чрез което мерим скоростта на нарастването, има стойност

$$\frac{p_{n;k}}{p_{n;k+1}} = \frac{(1 - p)^{n-k}}{(1 - p)^{n-k-1}} = 1 - p.$$

Разстоянието от това число до единицата е точно  $p$ .

Когато прилагаме търсене с връщане в задачата за цариците, тогава близо до точката на устойчиво равновесие (тоест за  $k \approx 0,5n$ ) важат следните оценки:

$$\begin{aligned} \frac{p_{n;k+1}}{p_{n;k}} &\approx \left( \frac{1-p \left(1 - \frac{k+1}{n}\right)^3}{1-p \left(1 - \frac{k}{n}\right)^3} \right)^n \approx \left( \frac{1 - \left(\frac{1}{2} - \frac{1}{n}\right)^3 p}{1 - \frac{1}{8}p} \right)^n = \\ &= \left( 1 + \frac{\left(\frac{3}{4n} - \frac{3}{2n^2} + \frac{1}{n^3}\right)p}{1 - \frac{1}{8}p} \right)^n = \left( 1 + \frac{3}{4} \cdot \frac{p}{1 - \frac{1}{8}p} \cdot \frac{1}{n} + \frac{p}{1 - \frac{1}{8}p} \cdot o\left(\frac{1}{n}\right) \right)^n. \end{aligned}$$

За големи  $n$  последният израз е приблизително равен на

$$\exp\left(\frac{3}{4} \cdot \frac{p}{1 - \frac{1}{8}p}\right),$$

а за плътности около критичната (следователно за  $p \approx 0$ )

$$\exp\left(\frac{3}{4} \cdot \frac{p}{1 - \frac{1}{8}p}\right) \approx 1 + \frac{3}{4} \cdot \frac{p}{1 - \frac{1}{8}p} \approx 1 + \frac{3}{4}p.$$

Щом

$$\frac{p_{n;k+1}}{p_{n;k}} \approx 1 + \frac{3}{4}p,$$

то

$$\frac{p_{n;k}}{p_{n;k+1}} \approx \frac{1}{1 + \frac{3}{4}p} \approx 1 - \frac{3}{4}p.$$

Последното приближение също важи за малки  $p$ , тоест за  $p \approx 0$ . Разстоянието от полученото частно до единицата е  $0,75p$ .

Тъй като  $0,75p < p$ , то формулата за стандартното отклонение е по-точна при търсене на разположение в задачата за цариците, отколкото при търсене на хамилтонов цикъл.

Забележка: През лятото на 2021 г. Майкъл Симкин публикува [10] следната асимптотична формула за броя на разположенията на  $n$  царици:

$$Q(n) \sim \left( (1 \pm o(1)) \frac{n}{e^\alpha} \right)^n, \text{ където } 1,94 < \alpha < 1,9449.$$

Статията с формулата и нейното доказателство е публикувана в arXiv, преди да е преминала през обстойна проверка от независим рецензент, тоест новият резултат все още чака одобрението на научната общност. В анализа на задачата за цариците приложихме формулата на Кльоатър. Ако вместо нея използваме формулата на Симкин, няма да се промени изразът за критичната плътност. Изводът на израза завършва така:

$$\frac{n! \left( p_{\text{кр.}} \right)^n}{c^n} \approx \ln 2, \text{ откъдето } p_{\text{кр.}} \approx \frac{c \sqrt[n]{\ln 2}}{\sqrt[n]{n!}} \sim \frac{c \cdot 1}{n/e} = \frac{ec}{n} \approx \frac{6,9}{n} \approx \frac{7}{n}.$$

Дробта  $\frac{n!}{c^n}$  в началото на горния ред е асимптотичната оценка на  $Q(n)$  според формулата на Кльоатър. Ако използваме формулата на Симкин, трябва да заменим  $\frac{n!}{c^n}$  със  $\left( (1 \pm o(1)) \frac{n}{e^\alpha} \right)^n$ , като вземем под внимание, че  $6,958 < e^\alpha < 6,993$ , тоест  $e^\alpha \approx 7$ . Следователно

$$\left( (1 \pm o(1)) \frac{n p_{\text{кр.}}}{e^\alpha} \right)^n \approx \ln 2, \text{ откъдето } p_{\text{кр.}} \approx \frac{e^\alpha \cdot \sqrt[n]{\ln 2}}{(1 \pm o(1))n} \sim \frac{e^\alpha}{n} \approx \frac{7}{n}.$$

С други думи, макар че формулата на Кльоатър и формулата на Симкин дават различни асимптотични оценки за  $Q(n)$ , все пак тази разлика не влияе на използваното от нас приближение на критичната плътност. Не се променят и останалите изводи от нашия анализ.

### 3 Судоку

Играта судоку е главоблъсканица, предназначена за един играч. Игралното поле е квадрат  $9 \times 9$ , разделен на девет квадрата  $3 \times 3$ . Единичните квадратчета трябва да се попълнят с цифрите от 1 до 9 така, че всяка цифра да се среща точно веднъж във всеки квадрат  $3 \times 3$ , както и във всеки ред и стълб на игралното поле. В началото на играта някои полета са попълнени.

2			5			8		3
		6		4	9			
5	1				2		4	9
4	6					9	1	5
			1		3			
9	2	1					8	7
8	3		4				6	2
			3	7		5		
6		7			8			

2	9	4	5	6	1	8	7	3
3	7	6	8	4	9	2	5	1
5	1	8	7	3	2	6	4	9
4	6	3	2	8	7	9	1	5
7	8	5	1	9	3	4	2	6
9	2	1	6	5	4	3	8	7
8	3	9	4	1	5	7	6	2
1	4	2	3	7	6	5	9	8
6	5	7	9	2	8	1	3	4

*Условието (отляво) и решението (отдясно) на едно судоку*

В зависимост от входните данни една задача може да няма решение, може да има единствено решение, може да има повече от едно решение. Задачите, давани по вестници и списания, традиционно притежават само едно решение.

Главоблъсканицата може да се решава по метода търсене с връщане. Избираме някое непопълнено квадратче и записваме в него някоя цифра, която не се среща в неговия ред и стълб, нито в неговия квадрат  $3 \times 3$ . Ако успеем да направим това, отиваме на друго непопълнено квадратче. В противен случай се връщаме на предишното попълнено квадратче и избираме друга цифра; ако няма такава, се връщаме с още една стъпка. В крайна сметка или успяваме да решим задачата, или установяваме, че тя няма решение.

Този алгоритъм е прекалено бавен — дотолкова, че в някои случаи отговорът изобщо не може да бъде дочакан.

Търсенето може да бъде ускорено чрез следните правила:

1) За всяко непопълнено квадратче пазим един списък от цифрите, с които можем да попълним квадратчето.

2) За всяка цифра от 1 до 9 и за всеки ред, стълб и квадрат  $3 \times 3$ , в които тази цифра не е попълнена, пазим списък от квадратчетата на реда, стълба или квадрата  $3 \times 3$ , които можем да попълним с нея.

3) На всяка стъпка търсим разбиване на най-малък брой случаи, тоест търсим един най-къс списък измежду списъците от т. 1 и т. 2. Ако не са останали списъци, т.е. ако всички квадратчета са попълнени, задачата е решена; отпечатваме решението и търсенето приключва.

4) Ако намереният списък е празен, тогава задачата няма решение при така направените избори на предишните стъпки. Следва връщане: променяме последния направен избор. Ако това се окаже невъзможно, тоест ако всички възможности са изчерпани, то задачата няма решение и търсенето приключва.

5) Ако намереният списък има единствен елемент, нямаме избор: попълваме цифрата в квадратчето и актуализираме другите списъци. Отиваме на т. 3.

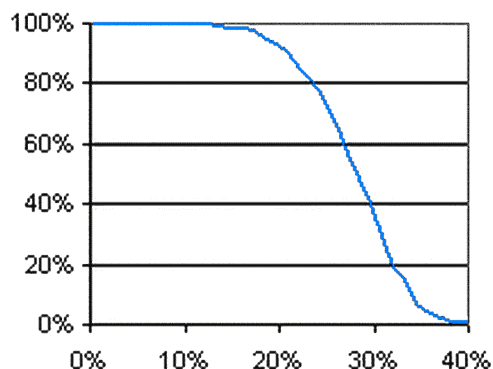
6) Ако намереният списък има няколко елемента, избираме един, попълваме цифрата в квадратчето и актуализираме другите списъци. Отиваме на т. 3. При връщане от т. 4 избираме друг елемент и т.н.

Важно е да се използват и двата вида списъци — от т. 1 и т. 2. Иначе алгоритъмът работи неприемливо бавно при някои входни данни. Както е описан по-горе, алгоритъмът работи изключително бързо при всякакви входни данни.

Алгоритмите за търсене от предишните глави бяха неоптимизирани. За първи път в това изследване разглеждаме оптимизиран алгоритъм. Възможно е да има разлики между него и неоптимизираните алгоритми. Такива разлики могат да бъдат открити опитно.

Алгоритъмът, описан по-горе в шест стъпки, беше програмиран и изпробван експериментално. Освен коментираното вече бързодействие той притежава и други интересни характеристики.

В играта sudoku също може да се въведе параметър плътност — това е абсолютният брой или относителната част на квадратчетата, попълнени в началото на играта (т.е. входните данни на алгоритъма). Ако тълкуваме като случайни разположения на тези квадратчета и попълнените в тях цифри, то отговорът на задачата е предвидим само с някаква вероятност. Експериментално получихме графиката на вероятността за съществуване на решение като функция на плътността.



По ординатата е нанесена вероятността за съществуване на решение, а по абсцисата — плътността на игралното поле.

Веднага се забелязва, че вероятността за съществуване на решение е намаляваща функция на плътността. Това е естествено да се очаква. Интересно е, че за повечето плътности вероятността е близо до 0 или 1, а забележимо различна от 0 и 1 тя е само в сравнително тесен участък около плътността 28%, съответстваща на 23 цифри, дадени отначало (при тази плътност вероятността за съществуване на решение е 0,5). Това свойство е присъщо на множество различни алгоритмични задачи; то се явява аналог на законите “0 или 1” от теорията на вероятностите. В други задачи го използвахме многократно за ускоряване на търсенето: извън критичния участък алгоритъмът връща отговор по подразбиране. В настоящата задача то е от по-малка полза, тъй като разполагаме с бърз алгоритъм, който няма нужда от по-нататъшно оптимизиране. Все пак се срещат, макар и рядко, входни данни, които изискват малко по-продължителна обработка от алгоритъма, така че си струва да анализираме подобни случаи.



Не е трудно да се изведе приближение за критичната плътност. Нека  $n$  е дължината на входа (броят на цифрите, дадени отначало). Докато  $n$  е малко (и цифрите се разполагат на случайни квадратчета), местоположенията на цифрите с голяма вероятност ще са независими, тоест ще лежат в различни редове, стълбове и квадрати  $3 \times 3$ . Затова първите няколко цифри няма да се ограничават взаимно и всяка следваща ще намалява броя на решенията около девет пъти. За големи  $n$  цифрите ще бъдат зависими и всяка следваща ще намалява броя на решенията по-малко, например осем пъти. Но за малки  $n$  делителят е около 9. Ако математическото очакване на броя решения е голямо число, то и вероятността за съществуване на решение е голяма. И обратно, ако математическото очакване е почти нула, толкова е и вероятността за съществуване на решение. Критичната плътност съответства на такова  $n$ , при което очакваният брой решения е малък, но не твърде близо до 0, т.е. от порядъка на единицата. Като вземем предвид, че при липса на ограничения съществуват [11]  $6670903752021072936960 \approx 6,67 \cdot 10^{21}$  решения, то

$$\frac{6,67 \cdot 10^{21}}{9^n} \approx 1 \iff n \approx \log_9(6,67 \cdot 10^{21}) = 22,87 \dots \approx 23,$$

което съответства на плътност  $23/81 \approx 28\%$ .

Това, че вероятността за съществуване на решение представлява намаляваща функция на плътността, е свойство на самата задача, не на алгоритъма.

Съществуват закономерности, които характеризират алгоритъма. Нека  $(\xi_t)_{t \geq 0}$  е целочислена редица, тълкувана като случаен процес: тя зависи от входните данни, а те могат да се смятат за случайни. Величината  $\xi_t$  е броят на квадратчетата, попълнени от алгоритъма след стъпка №  $t$ . Тази величина е пропорционална на количеството памет, нужно за поддържане на структура от данни, чрез която се осъществява връщането след неуспешна стъпка.

За този процес експериментите показват същото поведение, което забелязахме при другите задачи: траекториите се състоят от стръмно, но краткотрайно нарастване в началото на процеса, последвано от водоравен участък, в който траекторията се колебае около някаква средна стойност.

Поради бързината на алгоритъма почти няма дълги траектории. Само в 93 от 29000 теста дължините на траекториите надхвърлят 300, като цитираният брой (93) се отнася общо за всички плътности. Затова се налага да сравняваме входни данни с различна плътност, което не е желателно, но все пак води до някакви изводи.

Изследвахме средните стойности и стандартните отклонения на извадки от водоравните участъци на няколко дълги траектории:

— Когато задачата има решение:

Дължина $n$ на входните данни	3	4	8	12	16	20	24
Средна стойност	57,4	62,5	57,0	51,1	48,6	34,7	30,0
Средна стойност + $n$	60,4	66,5	65,0	63,1	64,6	54,7	54,0
Стандартно отклонение	4,8	3,9	5,9	4,9	6,5	8,3	5,2

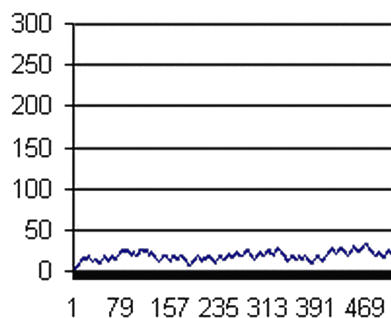
— Когато задачата няма решение:

Дължина $n$ на входните данни	20	21	21	23	23	25	25
Средна стойност	17,5	18,2	17,1	14,5	16,4	23,1	23,5
Средна стойност + $n$	37,5	39,2	38,1	37,5	39,4	48,1	48,5
Стандартно отклонение	3,0	2,9	6,6	3,6	7,3	4,7	7,2

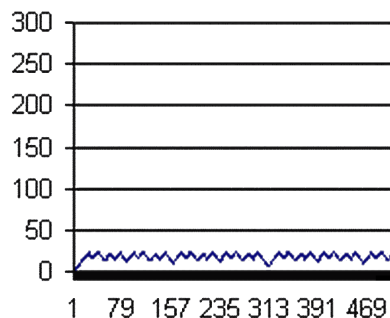
Отклоненията са близки, но средните стойности се различават ясно: те са отчетливо по-големи, когато задачата има решение. В този случай средните стойности намаляват с увеличаване на броя на дадените цифри, затова разпределението им се различава от нормалното разпределение и най-често използваните статистически формули стават неприложими. Но забелязаната закономерност и така е достатъчно ясна, поради което може да се използва за преждевременно прекъсване на дългите процеси, ако целта ни е единствено да разпознаем дали задачата има решение, а самото решение не ни интересува.

От друга страна, трябва все пак да се държи сметка за това, че в тази задача генералните съвкупности не са много добре отделени: за плътности около критичната разлика между средните стойности е сравнима със сбора на отклоненията и дори е по-малка от него, а това означава, че може да попаднем на траектории, за които е трудно да преценим от кой вид са.

На картинките са показани траектории на случайния процес  $(\xi_t)_{t \geq 0}$ , когато задачата има решение (отляво) и когато няма решение (отдясно). По абсцисната ос е нанесено дискретното време  $t$  на случайния процес, а по ординатната ос — количеството допълнителна памет в момента  $t$  (измерено чрез броя  $\xi_t$  на квадратчетата, попълнени от алгоритъма). Двете графики се отнасят за входни данни с еднаква дължина — 21 цифри, попълнени отначало.



Задачата има решение;  
 средна стойност: 16,5;  
 ср. кв. отклонение: 3,8.



Задачата няма решение;  
 средна стойност: 18,2;  
 ср. кв. отклонение: 3,1.

Не се забелязва съществена разлика между средните стойности, нито между отклоненията, в сравнение с разликите, показани по-горе. Извадките, съставени от членовете с индекси от 151 до 300 вкл., имат обем 150, средни стойности 16,5 и 18,2 и отклонения 3,8 и 3,1. Наистина, проверката на съответните статистически хипотези показва, че с много голяма сигурност ( $p$ -стойности  $< 0,007$ ) лявата траектория притежава по-голямо отклонение и по-малка средна стойност от дясната траектория. Обаче този извод трябва да бъде отхвърлен по много причини. Първо, малката  $p$ -стойност се дължи на големия обем, а така не спестяваме много време, ако спрем алгоритъма предсрочно. При извадки с по-малък обем потвърждението не е толкова сигурно. Второ, посоката на неравенството между средните стойности е обратна на установеното по таблиците по-горе. Ако изобщо прекъсваме търсенето преди естествения му край, би трябвало да работим с неравенството, установено от таблиците; в такъв случай поне едната от двете траектории ще заблуди разпознавателя.

Такъв тип грешки са неизбежни: двете генерални съвкупности не са достатъчно добре отделени (интервалите им се припокриват), поради което разпознаването с предсрочно прекратяване на търсенето е възможно (със съответна печалба на време), но е доста рисковано. При наличието на бърз алгоритъм не е препоръчително да се използва предсрочно прекратяване на търсенето: печалбата на време е малка, а загубата на сигурност е голяма.

## Заклучение

В настоящата дипломна работа изследвахме няколко алгоритъма, построени по схемата търсене с връщане, решаващи различни задачи, и някои числови характеристики на изменението на количеството памет, използвана от алгоритъма, като приложихме методи от статистиката, тълкувайки изменението на количеството памет като случаен процес. При някои алгоритми можем отрано да прогнозираме резултата и да спестим част от времето, като прекъснем алгоритъма предсрочно. Ако в задачата не се допуска такова прекъсване, то във всички случаи може да се предвиди типичното количество памет, нужно на алгоритъма, и да се планират характеристиките на средата, в която ще се изпълнява. Надеждността на прогнозите зависи от вероятностни разпределения, свързани с конкретната алгоритмична задача, и може да се установи както експериментално, така и чрез теоретични разсъждения.

Разни закономерности, забелязани по време на експериментите, получиха обяснение въз основа на разработените теоретични модели. Въпреки че има разлики в конкретните подробности на всяка задача, използваният подход се оказва достатъчно общ, така че обхваща практически всички задачи от разглеждания вид. Основната идея е търсенето с връщане да се тълкува като симетрично случайно лутане в околност на точката на устойчиво равновесие.

Самата идея за откриване на корелации между изхода на алгоритъм и някои числови характеристики на изменението на количеството памет, използвано от алгоритъм, построен по схемата търсене с връщане, както и прилагането на такива корелации за ускоряване на алгоритъма принадлежат на научния ръководител на дипломната работа ([12], [13]).

Експерименталните резултати са получени заедно от дипломанта и научния ръководител.

Теоретичният анализ и обосновка на експерименталните резултати са самостоятелен принос на дипломанта.

Въпреки приликите в прилагането на подхода към различни задачи все още не са известни теоретични обобщения, които биха позволили прогнозирането на типичното количество памет за търсене с връщане въз основа на често срещани и лесно разпознаваеми характеристики на съответната алгоритмична задача. Откриването на такива обобщения е подходяща насока за бъдещи изследвания в тази научна област.

Макар че предложеният стохастичен модел на търсенето с връщане може да бъде прилаган поначало към всеки алгоритъм от този вид, все пак успешното му прилагане към конкретна алгоритмична задача изисква наличието на разработена теория за нея, която съдържа достатъчно точни оценки на важни числови характеристики на задачата (съответстващи на понятието критична плътност, използвано от модела). Възможностите за прогнозиране и оптимизиране, предлагани от модела, са стимул за развитието на теоретичните познания за онези задачи, които могат да се решават чрез търсене с връщане.

## Литература

- [1] Бела Болобаш, *Теория на графите*, “Наука и изкуство”, София, 1989.
- [2] Lajos Pósa, *Hamiltonian circuits in random graphs*, Discrete Mathematics, 1976, volume 14, issue 4, pages 359–364, <https://doi.org/10.1016%2F0012-365X%2876%2990068-6>.
- [3] A. Korshunov, *Solution of a problem of Erdős and Rényi on Hamilton cycles in non-oriented graphs*, Soviet Math. Dokl. **17**, 1976, 760–764.
- [4] Д. Дончев, М. Дилчева, В. Кинова, *Практическо ръководство по статистика*, “Автоспектър”, Пловдив, 2002.
- [5] А. Н. Ширяев, *Вероятностъ*, “Наука”, Москва, 1980.
- [6] Й. Стоянов, И. Миразчийски, Ц. Игнатов, М. Танушев, *Ръководство по теория на вероятностите*, Университетско издателство “Св. Климент Охридски”, София, 1991.
- [7] *Eight queens puzzle*, Wikipedia, [https://en.wikipedia.org/wiki/Eight\\_queens\\_puzzle](https://en.wikipedia.org/wiki/Eight_queens_puzzle).
- [8] N. J. A. Sloane, *The On-Line Encyclopedia of Integer Sequences*, <https://oeis.org/A000170>.
- [9] Jordan Bell, Brett Stevens, *A survey of known results and research areas for n-queens*, Discrete Mathematics, volume 309, issue 1, 6 January 2009, pages 1–31; <http://dx.doi.org/10.1016/j.disc.2007.12.043>.
- [10] Michael Simkin, *The number of n-queens configurations*, 2021, <https://arxiv.org/abs/2107.13460v2>.
- [11] *Mathematics of Sudoku*, Wikipedia, [https://en.wikipedia.org/wiki/Mathematics\\_of\\_Sudoku](https://en.wikipedia.org/wiki/Mathematics_of_Sudoku).
- [12] D. Kralchev, *Memory consumption in NP-complete problems*, Пролетна научна сесия на ФМИ на СУ — 31 март 2018 г.
- [13] Д. Кралчев, *Статистически хипотези за алгоритми*, Пролетна научна сесия на ФМИ на СУ — 27 март 2021 г.