Gmail

Maciek Pct <maciek.pacut@gmail.com>

Zysk na S*

29 messages

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Wed, Feb 19, 2020 at 9:04 PM
To: maciek <maciek.pacut@gmail.com>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Cześć

Tak sobie na spokojnie już myślałem o knapsacku i mam taki pomysł na uproszczenie. (Nie wiem, co by przypadło bardziej do gustu recenzentom, ale chyba w finalnej wersji albo w wersji czasopismowej warto byłoby takie uproszczenie wprowadzić).

Mianowicie:

- Zmienmy granice między małym i średnim przedmiotem z Phi = 0,21 na 1/4.
- Lepiej stackujmy małe. Obecnie robimy to na chama i wtedy gwarantujemy poziom w kubelkach S* rowny 1-Phi. Jakbyśmy je składali według rozmiarów, to moglibyśmy zagwarantować co najmniej 4/5 (wiec lepiej niż obecnie).

Nie sprawdzałem reszty, ale może widzicie od razu ze to nie zadziała?

M.

---

maciek <maciek.pacut@gmail.com> Wed, Feb 19, 2020 at 9:10 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Mielibyśmy wtedy dwa typy kubełków na małe: (0, 1/5] i (1/5, 1/4]. Dobrze rozumiem?

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Wed, Feb 19, 2020 at 9:11 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, maciek <maciek.pacut@gmail.com>

Dwa by chyba wystarczyły, ale tez na pewno dwa byłyby konieczne.

M.
[Quoted text hidden]

---

maciek <maciek.pacut@gmail.com> Wed, Feb 19, 2020 at 9:16 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Wydaje mi się, że to nie działa. Nawet z tą modyfikacją musimy scalać małe do średnich. Popatrz na Property 5 of Lemma 7 i odpowiadający mu przypadek w analizie. To przypadek ze scalonym medium. Z modyfikacją taki medium mógłby być większy niż 2\phi. Wszystkich constraintów dla pełnego przedziału (tzn. pełny = (phi, 1/2] a ograniczony = (phi, 2phi)) i kluczowe w analizie było to, żeby któryś rozluźnić.

[Quoted text hidden]

---

maciek <maciek.pacut@gmail.com> Wed, Feb 19, 2020 at 9:17 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

* wszystkich constraintów dla pełnego przedziału nie możemy spełnić jednocześnie i kluczowe (...)

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Wed, Feb 19, 2020 at 9:24 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, maciek <maciek.pacut@gmail.com>

Moment. Trochę już nie pamietam, co sobie myślałem, ale chyba nie chce scalać małych do średnich w ogóle.

M.
On 19 Feb 2020, 20:22 +0000, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, wrote:

> Wydaje mi się, że tam jest taki problem, że dwa elementy o rozmiarach 1/4 tworzą medium 1/2. A medium 1/2 z definicji nie mieści się w D. To oznacza, że odrzucamy go ze struktury i nie mamy argumentu, że taki element w strukturze już istnieje. Przez co zysk wydaje mi się wynosić P(4/5) < R. Musiałbym to lepiej przemyśleć.
>
> On Wed, Feb 19, 2020 at 9:13 PM Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> wrote:
>
>> Tak jak obecnie M* / tak jak Maciek napisał.
>>
>> M.
>> On 19 Feb 2020, 20:12 +0000, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, wrote:
>>
>>> Co to znaczy "według rozmiarów"?
>>> [Quoted text hidden]

---

maciek <maciek.pacut@gmail.com> Wed, Feb 19, 2020 at 9:42 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

To byłaby znaczące uproszczenie algorytmu! Fajnie jakby zadziałało :)

[Quoted text hidden]

---

maciek <maciek.pacut@gmail.com> Thu, Feb 20, 2020 at 12:10 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Uważam, że mamy małe szanse na pozbycie się scalania małych z algorytmu. Nawet gdyby:
- ograniczyć input do jednego rozmiaru małych oraz przedmiotów >1/3
- założyć zysk z S* dowolnie bliski 1 (równy np. 0.999)
to wciąż mielibyśmy zysk P(0.999) < R w przypadku gdy najpierw przyjdzie f^{-1}(0.999) dużych, a następnie małe do zapełnienia kubełków.

Wyrażę też opinię, że rozwiązanie kwestii małych było prawdziwym polem minowym, gdyż trzeba było spełnić na raz wiele warunków. Od początku 2016 roku do grudnia 2017 nawet się poddaliśmy i spisywaliśmy wersję 0.5779-konkurencyjną.

Nie uważam, że nie da się tego zrobić. Ale gdy chcemy rozszerzać algorytm o małe, bez zmian dla średnich i dużych, to raczej dojdziemy do aktualnego rozwiązania. Potrzebne byłoby jednak zupełnie nowe podejście, np. jakieś generalizacje związku D z M* (i wtedy pewnie też S*). Swojego czasu spędziłem dużo czasu nad takimi pomysłami i chętnie bym to pociągnął gdybyśmy mieli jakiś punkt zaczepienia.

Maciek

[Quoted text hidden]

---

maciek <maciek.pacut@gmail.com> Thu, Feb 20, 2020 at 12:51 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

*mielibyśmy zysk P(0.999) < R w przypadku gdy najpierw przyjdą małe i zapełnią 1-f^{-1}(0.999) kubełków, a następnie duże po thresholdzie

sorry :)

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Thu, Feb 20, 2020 at 4:53 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, maciek <maciek.pacut@gmail.com>

O, to bardzo ładna konstrukcja, dzieki!

M.
[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Thu, Feb 20, 2020 at 6:53 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, maciek <maciek.pacut@gmail.com>

Chyba nie rozumiem tego argumentu :(

M.
On 19 Feb 2020, 20:22 +0000, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, wrote:

> Wydaje mi się, że tam jest taki problem, że dwa elementy o rozmiarach 1/4 tworzą medium 1/2. A medium 1/2 z definicji nie mieści się w D. To oznacza, że odrzucamy go ze struktury i nie mamy argumentu, że taki element w strukturze już istnieje. Przez co zysk wydaje mi się wynosić P(4/5) < R. Musiałbym to lepiej przemyśleć.
>
> On Wed, Feb 19, 2020 at 9:13 PM Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> wrote:
>
>> Tak jak obecnie M* / tak jak Maciek napisał.
>>
>> M.
>> On 19 Feb 2020, 20:12 +0000, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, wrote:
>>
>>> Co to znaczy "według rozmiarów"?
>>>
>>> On Wed, Feb 19, 2020 at 9:05 PM Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> wrote:
>>> [Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Thu, Feb 20, 2020 at 7:04 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, maciek <maciek.pacut@gmail.com>

Przekonałeś już mnie, ze trzeba scalać małe do średnich, bo bez tego na pewno nie działa.

Ale poniższy argument mnie chyba nie przekonuje. Nie przeliczyłem tego, ale powinnismy mieć wtedy więcej luzu, bo w takim przypadku level(D) = 4/5, co jest więcej niż do tej pory, wiec może tu się da coś uzyskać?

Jak będę miał dostęp do komputera / papieru to sprobuje to przeliczyć.

M.
[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Thu, Feb 20, 2020 at 7:32 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Rozważmy taką sekwencję wejściową: przychodzą małe o rozmiarze 1/4 i zajmują 1-f^{-1}(4/5) kubełków S*. Scalamy 1/4 od razu do 1/2 i nie mamy D, bo \xi(1/2) = 0. Teraz przychodzą duże po thresholdzie. Mamy zysk P(4/5) < R.

\xi(1/2) musi wynosić 0, bo mamy górne ograniczenie na \xi, które schodzi do 0 przy 1/2.

Poprawiłeś zysk na klasie S* z 1-\phi do 4/5. Póki co jedyne, co ja umiem z tym zrobić, to podstawienie (1-\phi) na 4/5 w Eq. 7.3 i 7.5 i pewnie fajniejszy ich dowód. Nie umiem zredukować liczby przypadków ani zmienić algorytmu.
[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Thu, Feb 20, 2020 at 9:03 PM
To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, Maciek <maciek.pacut@gmail.com>

Kolejny bardzo fajny przykład :))

Pewnie dałoby się z nich zrobić kiedyś jakaś narracje do intro pt. „Dlaczego nasz algorytm nie może być prostszy” :)

W każdym razie to może nawet lepiej. Proponuje, ze potem sprobuje dodać w nowym branchu klasyfikacje małych: może się uproszczą jakieś dowody w appendiksie.

M.

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Fri, Feb 21, 2020 at 10:07 AM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Chcielibyśmy zrobić algorytm regularny, czyli taki, który może mieć granice klas przedmiotów tylko w 1/i. Wczoraj podałem sekwencję, która miała przekonywać, że się nie da, a dzisiaj podaję algorytm i analizę, która sobie z tą sekwencją radzi.

Dobrze znany nam algorytm modyfikuję jedynie przesuwając granicę między małymi a średnimi do 1/4.
Ograniczmy wejście do przedmiotów {z}u(1/4, 1], gdzie z \in (1/5, 1/4] jest pojedynczym małym przedmiotem.

Jeśli mamy kubełki S*, to w D nie mieści się przedmiot 2z, co oznacza, że 2z lub mniejszy jest ciasny, co daje gain(D) > 2z*\xi(2z). Zysk na kubełkach S* to 4z.

Załóżmy, że 4z < pile(mt(D)). Sumaryczny zysk algorytmu to

P(4z)+Q(1-x)+(x-4z)\xi(x)+T(x,2z) dla dowolnego 1/3<x<2z<1/2

Na szczęście udowodniliśmy już, że powyższe jest wyższe niż R w Eq. 7.6, gdyż pile(2z)=4z.

Ciekawe, czy da się to uogólnić do przedziału. Powyższe jest dla 1 rozmiaru przedmiotu.

Maciek

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Fri, Feb 21, 2020 at 10:44 AM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

> Ciekawe, czy da się to uogólnić do przedziału.

Raczej się da.

Jednym kandydatem byłoby 2z := "rozmiar najmniejszego scalonego odrzuconego ze struktury". To gwarantuje zysk 4z na M* i 2z odrzucalne ze struktury (a z monotoniczności \xi również zysk 2z*\xi(2z) na D).

Ale zastanawiam się, czy nie dałoby się prościej, tzn. podpiąć pod istniejącą analizę średnich.

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Fri, Feb 21, 2020 at 3:10 PM

To: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>, Maciek <maciek.pacut@gmail.com>

Czy dobrze rozumiem, ze w skrócie to co chcesz tutaj powiedzieć to ze Twój wczorajszy przykład mówi po prostu ze nie można poprzestać na mówieniu ze zysk na S* to 4/5 tylko trzeba analizować ten zysk dokładniej?

M.

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Fri, Feb 21, 2020 at 3:49 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Tak, szacowanie 4/5 na S* nie wystarczy i tak można interpretować ten przykład. Ale dla przedziału (1/5, 1/4] możemy analizować zysk dokładniej.

Dla pozostałych małych (0, 1/5] musimy zadowolić się pewnym stałym zyskiem, np. 4/5 (lub jeśli to coś da, wydzielić kolejną klasę (1/6, 1/5]).

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Fri, Feb 21, 2020 at 10:22 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Chyba potrafię uprościć analizę naszego algorytmu. Cel: algorytm regularny (granice między klasami tylko w 1/i), którego cała analiza dla przedmiotów (0, 1/2] przebiega tak, jak analiza średnich. Bez osobnego przypadku, gdzie zysk S* = jakaś stała. Małe = (0, 1/4].

> Dla pozostałych małych (0, 1/5] musimy zadowolić się pewnym stałym zyskiem, np. 4/5 (**lub jeśli to coś da, wydzielić kolejną klasę (1/6, 1/5]**).

Okazuje się, że to jest dobry pomysł. Wydzielmy jeszcze tę jedną podklasę małych i zbierajmy przedmioty z niej osobno. Tzn. dzielimy małe na A = (0, 1/6], B = (1/6, 1/5] i C = (1/5, 1/4] i zbierajmy małe do 3 różnych typów kubełków: A, B i C. Jeśli load w którymś z nich przekracza 1/4 to mogą się scalić w M --- chyba że się nie mieszczą w D, to stają się odpowiednio A*, B*, C*.

Analiza jak dla średnich: Dla przedmiotów B (i odpowiednio C) możemy powiązać zysk na B* (odpowiednio C*) na minimalnym odrzuconym scalonym z klasy B (odpowiednio C) tak jak napisałem w analizie "wczorajszego" algorytmu: gdy zysk na C* wyniesie 2z, mamy minimalny ciasny w strukturze co najwyżej równy z. I dla B* jest tak samo.

Pozostaje pytanie, jak powiązać zysk na A* z minimalnym odrzuconym scalonym z klasy A w taki sam sposób (2z vs z). Otóż dla tej klasy jest to bardzo proste (i dopiero dla tej -- dla B i C nie działa). Jak duży może być scalony przedmiot w kubełku A? Co najwyżej 1/4 + 1/6 ~= 0.41. A zysk na A* to przynajmniej 1-1/6 = 0.83. Czyli mamy własność, że zysk na A* to dwukrotność zysku na minimalnym odrzuconym scalonym z A. Tak jak w analizie średnich.

Czytajcie, sprawdzajcie.

Maciek

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Fri, Feb 21, 2020 at 11:41 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>, Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

> I dla B* jest tak samo.

Argument jest taki sam jak dla C/C*, bo też scalają się po 2, bo już 2x B przekraczają 1/4. Czyli scalone B to co najwyżej 2/5=0.4, a B* to co najmniej 1-1/5=0.8.

Maciek

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Sun, Feb 23, 2020 at 5:47 PM
To: Maciek <maciek.pacut@gmail.com>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Wygląda sensownie! :)
Czy dobrze rozumiem, że w efekcie powinna się uprościć analiza funkcji \xi?

W ciągu najbliższych dwóch tygodni nie będę miał czasu, żeby się tym zajmować. Jeśli masz czas, to możesz zrobić kopię katalogu ICALP-submitted i tam działać.

PS. I chyba katalog OLD możemy już wysłać na emeryturę i skasować? :)

M.
[Quoted text hidden]
--
Marcin Bieńkowski
Combinatorial Optimization Group
Institute of Computer Science, University of Wrocław

Maciek <maciek.pacut@gmail.com> Sun, Feb 23, 2020 at 7:50 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Tak, analiza powinna się znacząco uprościć. Funkcja \xi powinna mieć tylko 4 własności. Cała analiza małych zostanie zastąpiona przez jeden lemat (tj. w treści poprzedniego maila) i zmianę w algorytmie. Nie będzie funkcji level. Jeśli średnie są >1/4 to mamy np. własność: 1-m < 1/2+m' dla średnich m,m'. To tak na pierwszy rzut oka, a może uproszczenia pociągną następne.

Marcinie, jak będziesz miał kiedyś chwilę to proszę napisz, co uważasz, że mógłbym zrobić do journala: rozwinąć, usunąć, przepisać, może coś przemyśleć. Tak, żeby można było napisać "Preliminary version of this paper was published at <hopefully ICALP>, and this paper is a thoroughly rewritten and improved version of it". I najlepiej tak, żeby była szansa na akceptację w jakimś journalu z CORE A* :)))

Póki co zajmę się tym, co mam na liście TODO:
1. Opis, dlaczego algorytm musi mieć te wszystkie zawiłości. Przedstawienie idei konstrukcji algorytmu polegającej na rozszerzaniu o jedną klasę 1/i na raz. Jaki input jest niebezpieczny, gdy zrobi się to naiwnie i jakie warunki są potrzebne, żeby nie zepsuć poprzednich klas.
2. Uproszczenie analizy małych.

Wishlist poprawek:
1. Czy możemy jakoś ominąć analizę dwuargumentową? Nawet kosztem jakiejś zmiany w algorytmie. A jeśli się nie uda, to czy dałoby się to ładniej sformalizować.

Usunąłem OLD, choć zostawiłem samego knapsack_OLD.pdf i progs/. Utworzyłem katalog journal. Będę tam dłubał w wolnych chwilach.

Maciek

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Mon, Feb 24, 2020 at 2:11 AM
To: Maciek <maciek.pacut@gmail.com>

Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Myślę, że jeśli będzie ładna analiza dla małych i spójną ta z analizą dla średnich, to punkt pierwszy (opis dlaczego algorytm musi mieć takie zawiłości) nie jest już taki potrzebny. To chyba byłaby najważniejsza zmiana.

M.

[Quoted text hidden]

Maciek <maciek.pacut@gmail.com> Mon, Feb 24, 2020 at 9:23 PM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

To podejście nie działa, tzn. trochę poprawia, ale nie wystarczająco do ujednolicenia analizy. Może za to przybliżamy się do przekonywującego argumentu za tym, że małe trzeba analizować osobno? Albo do argumentu za tym, że trzeba dzielić klasy poza punktami 1/i.

Popatrzmy na klasę (1/5, 1/4]. Zbieramy dwa pierwsze przedmioty i sprawdzamy czy się mieści. Jeśli się nie mieści, to mamy coś w strukturze. Ale nie mamy tej samej gwarancji co do dwóch kolejnych przedmiotów dokładanych do S*. Mogą przyjść minimalne i zysk jest wtedy mt(D)+2/5, co może być mniejsze od 2*mt(D).

Podział na dodatkowe klasy daje lepszy zysk na S* niż stała. Ale wciąż mniejszy niż 2*mt(D).

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Mon, Mar 2, 2020 at 10:31 PM
To: Maciek <maciek.pacut@gmail.com>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Rozumiem, że nie da się powtórzyć tego samego dowodu co dla średnich. Ale czy w tym przypadku co napisałeś, jest ogólnie źle, czy po prostu nasza obecna analiza tego nie łapie?

W każdym razie, pewnie warto kiedyś dodać lepsze składanie małych elementów, bo pewnie jakiś jeden czy dwa argumenty z analizą zmiennych w appendiksie się uproszczą?

M.
[Quoted text hidden]

Maciej Pacut <maciek.pacut@gmail.com> Tue, Mar 3, 2020 at 9:22 AM
To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Tak, jedynie argumentowałem, dlaczego nie da się powtórzyć dowodu dla średnich. Twoja sugestia składowania małych osobno i szacowania zysku na S* przez 4/5 działa i w znaczący sposób uprości analizę. Zajmę się wprowadzeniem tej modyfikacji.

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Tue, Mar 3, 2020 at 5:04 PM
To: Maciej Pacut <maciek.pacut@gmail.com>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

OK, świetnie. Tylko w innym katalogu niż ICALP-submitted. Może być w journal :-)

M.
[Quoted text hidden]

Maciej Pacut <maciek.pacut@gmail.com> Wed, Mar 4, 2020 at 11:49 AM

To: Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Jednak nie widzę jak Twoja sugestia może uprościć analizę.

Nie możemy rozszerzyć S do (0,1/4]. Już to wcześniej przedyskutowaliśmy w tym wątku, nie osiągniemy R w jednym z przypadków analizy. Mogę powtórzyć ten argument i możemy go dokładniej przedyskutować.

Zgodnie z Twoją sugestią, na kubełkach S* możemy mieć zysk 4/5: składając osobno (0, 1/4] i (1/4, \phi]. Pozostaje pytanie, czy warto w analizie podmienić 1-\phi na 4/5? To niczego nie uprości, w szczególności \phi wciąż pojawi się w wielu miejscach nawet w properties of \xi.

[Quoted text hidden]

Marcin Bieńkowski <marcin.bienkowski@cs.uni.wroc.pl> Wed, Mar 4, 2020 at 2:00 PM
To: Maciej Pacut <maciek.pacut@gmail.com>
Cc: Krzysztof Piecuch <krzysztof.piecuch@gmail.com>

Hm, potem się temu przyjrzę dokładniej.

Myślałem, że to może poprawić jakiś analityczny dowód jakiegoś property of \xi, bo będziemy mieć lepsze gwarancje na zysk algorytmu. Ale nie sprawdzałem dokładnie.

M.
[Quoted text hidden]