

## ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Харківський національний університет радіоелектроніки  
Кафедра ЕОМ

## Методи обробки даних в хмарних системах

Кваліфікаційна робота  
Другий (магістерський) рівень

Автор:  
Козін М.Д.,  
студ. гр. СПм-23-3

Керівник:  
Волк М.О.,  
проф. каф. ЕОМ

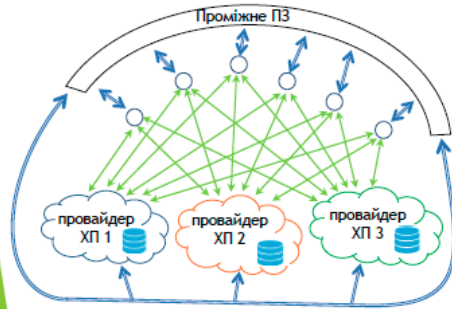
## Мета і задачі роботи

**Мета:** розробити метод узгодження вхідних даних у георозподілених репліках мультимарних систем.

### Задачі:

- ▶ визначити ефективні механізми реплікації вхідних запитів у георозподілених мультимарних системах
- ▶ розробити механізм узгодження вхідних запитів у мультимарних системах за номерами послідовності загального порядку
- ▶ розробити програмне забезпечення для верифікації розроблених алгоритмів

## Методи узгодження вхідних пакетів у репліках хмарних систем



Провайдери ХП у МХС з різними моделями узгодженості даних, тому ефективність МХС визначається затримкою відгуку, тобто, за зменшенням часу відповіді при запитах до георозподілених реплік МХС.

Основні недоліки при реалізації у МХС:

**протоколів реплікації кінцевого автомату** (Raft, Zab, Weave, CURP та численні варіанти Paxos: MultiPaxos, NOPaxos, Mencius, FastPaxos, Domino, Epochos, GeoPaxos+, PigPaxos)

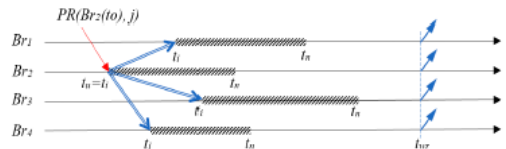
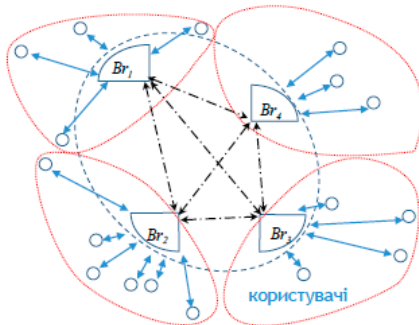
- необхідність кворуму для визначення репліки-лідера вимагає додаткових кроків глобального зв'язку
- проблеми з масштабованістю

**алгоритмічних методів реплікації на мітках часу**

- вимагають додаткових протоколів глобальної синхронізації та стабілізації годинників, що збільшує затримку відгуку

3

## Затримка узгодження вхідних пакетів у георозподілених репліках мультихмарних систем

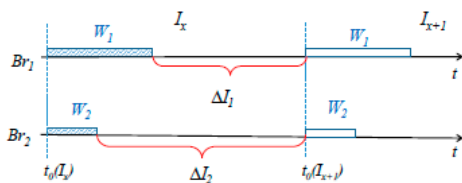


Затримка узгодження номерів вхідних пакетів у послідовність загального порядку:

$$Lat_{consist}(Br_x(to), j) = t_n(Br_x(to)) - t_u(Br_a(from))$$

4

## Інтервальне впорядкування вхідних пакетів



Вікно доступності  $W_x \geq$  середній час доставки пакетів від користувачів кластеру до свого  $Br_x$ .

Залишкове вікно  $\Delta I_x \geq$  середній час доставки пакетів від активного брокера до найвіддаленішого брокера МХС.

Єдине для усіх брокерів значення інтервалу коригування номерів вхідних пакетів:

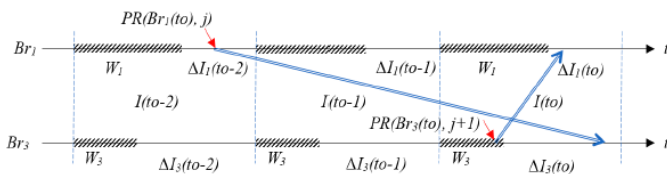
$$I = \max_{x=1..BrMax} I_x = \max_{x=1..BrMax} (W_x + \Delta I_x)$$

Термін інтервальної корекції номерів вхідних пакетів:

$$MaxLate = \max_{a,x=1..BrMax} t_{Br}(Br_a(from), Br_x(to)) = K_{Late} \cdot I$$

5

## Приклад формування кортежу вікон $M_{late}$ для коригування номерів з масиву пакетів $M_{cor}$



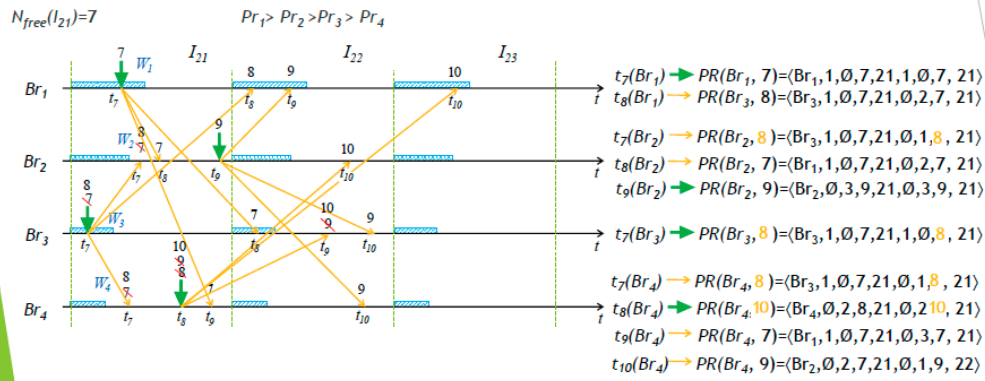
Якщо апріорно визначений термін інтервальної корекції вхідних пакетів  $MaxLate = 2 \cdot I$ , то кортеж вікон для впорядкування номерів вхідних пакетів:

$$M_{late} = \langle \Delta I_1(to - 2), W_1(to - 1), \Delta I_1(to - 1), W_1(to), \Delta I_1(to) \rangle$$

$M_{cor}$  містить усі пакети, що надійшли протягом вікон з кортежу  $|M_{late}| = 5$ .

6

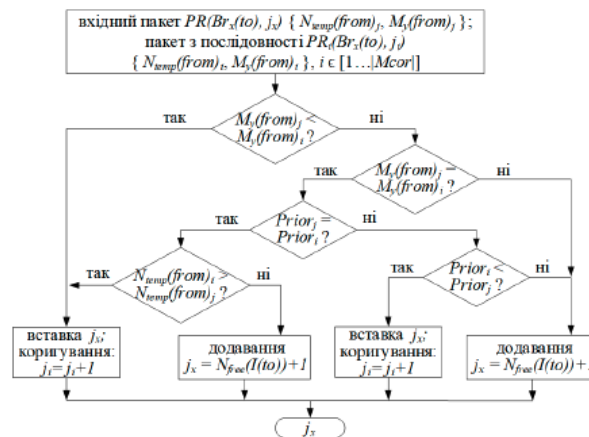
## Приклад впорядкування номерів пакетів за алгоритмом Corr



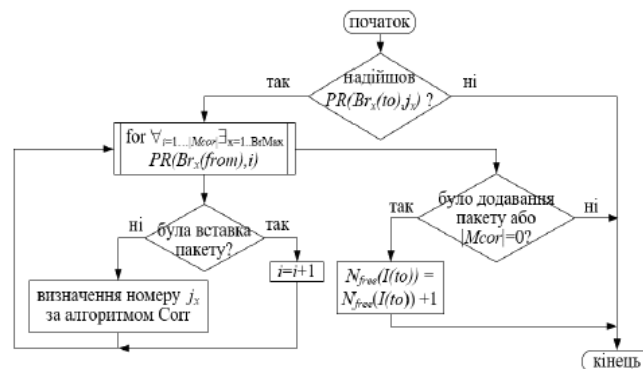
структура ключа-ідентифікатора вхідного пакету:

$$PR(Br_x(to), N_{gl}(to)) = (Br_x(from), N_{temp}^W(from), N_{temp}(from), N_{gl}(from), I(from), N_{temp}^W(to), N_{temp}(to), N_{gl}(to), I(to))$$

## Алгоритм Corr визначення номеру $j_x$ вхідного пакету у загальній послідовності

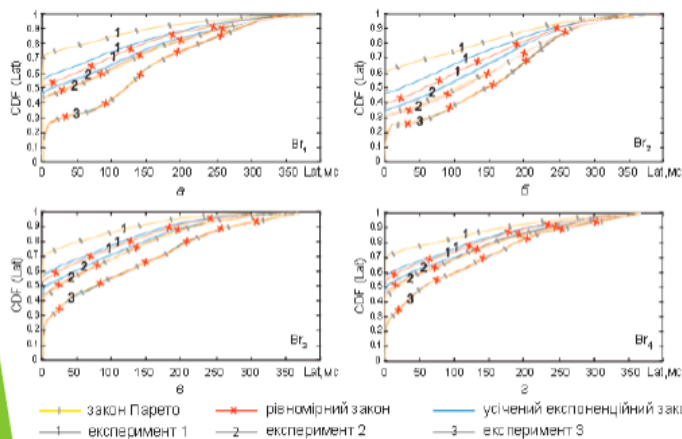


## Алгоритм узгодження номерів вхідних пакетів серед пакетів масиву коректування $Mcor$



## Результати експериментів 1-3

кумулятивна функція розподілу ймовірності  $CDF(Lat)$   
затримки формування впорядкованого номеру загальної послідовності



1,2,3: по три вхідні потоки до кожного брокера за закон Парето, усіченим експоненціальним та рівномірним законами

експеримент	середнє значення інтервалів між вхідними пакетами від користувачів, мс			
	до Br1	до Br2	до Br3	до Br4
1	148	97	163	112
2	37	24	41	28
3	2	2	2	2

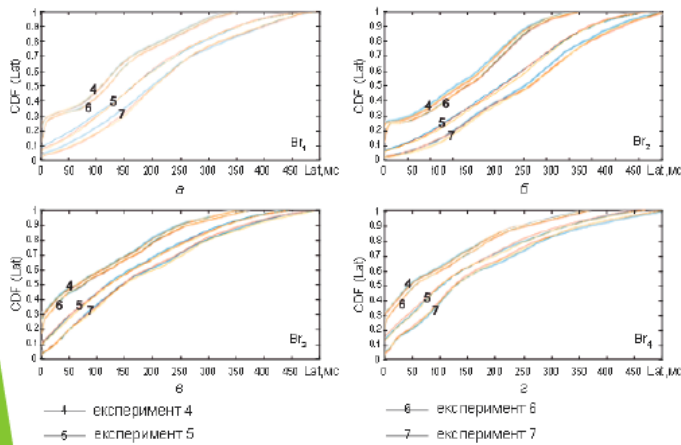
  

	Br <sub>1</sub>	Br <sub>2</sub>	Br <sub>3</sub>	Br <sub>4</sub>
Br <sub>1</sub>	0	8.0623	8.1048	8.1070
Br <sub>2</sub>	8.0425	0	8.1028	8.1103
Br <sub>3</sub>	8.1041	8.0480	0	8.0603
Br <sub>4</sub>	8.0988	8.1221	8.0839	0

рівномірний розподіл щільності часу доставки пакетів між брокерами

## Результати експериментів 4-7

кумулятивна функція розподілу ймовірності  $CDF(Lat)$   
затримки формування впорядкованого номеру загальної послідовності



4,5,6,7: по три вхідних потоки до кожного брокера за законам Парето, усіченим експоненціальним та рівномірним законами

4,6

	$\lambda_{r1}$	$\lambda_{r2}$	$\lambda_{r3}$	$\lambda_{r4}$
$\lambda_{r1}$	0	3.4552	1.7367	1.7372
$\lambda_{r2}$	3.4468	0	2.3151	3.1861
$\lambda_{r3}$	2.3154	2.2994	0	1.7272
$\lambda_{r4}$	1.7354	2.9007	1.7322	0

рівномірний розподіл щільності інтервалів доставки пакетів між брокерами

5,7

	$\lambda_{r1}$	$\lambda_{r2}$	$\lambda_{r3}$	$\lambda_{r4}$
$\lambda_{r1}$	0	57.0506	19.3489	10.0041
$\lambda_{r2}$	57.1512	0	50.1610	41.3886
$\lambda_{r3}$	19.3230	49.9747	0	47.1434
$\lambda_{r4}$	10.0319	41.4167	47.3009	0

усічений експоненціальний розподіл щільності інтервалів доставки пакетів між брокерами

11

## Висновки

- Згідно завдання в роботі визначена ефективність алгоритмічних методів реплікації вхідних запитів у георозподілених мультимарних системах.
- Розроблено алгоритм Согг корекції номерів вхідних пакетів у послідовності загального порядку. Розроблено метод впорядкування вхідних пакетів протягом апріорно визначених рівно тривалих непересічних інтервалів за алгоритмом Согг.
- Розроблено програмне забезпечення для верифікації розроблених алгоритмів. Отримані результати дозволяють не тільки знизити затримку запису даних у репліки вже існуючих мультимарних системах, але й обрати найкраще географічне розташування ресурсів постачальників хмарних послуг під час проектування нових.
- Результати роботи наведено у статтях:
  - Метод синхронізації запитів на запис даних у федеративних хмарних системах / Волк М.О., Козіна О.А., Козін М.Д. // Вісник НТУ "ХПІ". Серія: Інформатика та моделювання. - Харків: НТУ "ХПІ". - 2025. - № 1 (13). - С. 80 - 98 (опубліковано)
  - Development of a method for data consistency in replication in multicloud systems / Volk, M., Kozina, O., Buhrii, A., Osiievskiyi, S., Kozin, M., Volk, D., Diachenko, D., Turinskyi, Y. // Eastern-European Journal of Enterprise Technologies. 2025. No.4(2 (136)). P. 1-24 (Scopus, прийнята до публікації у серпневому номері 2025р.)

## ДОДАТОК Б

## Лістинг модулів імітаційної моделі

## Б.1 Файл laws3.m

```

clear, clc
RpMax = 50000; t0=10;
BRnum=[1 2 3 4 ]; % общее количество брокеров
rng(10, 'twister');

%% генерация потоков от юзеров
MeanUserToBr = [2 2 2 2];
%% экспоненциальное распределение интервалов между приходом пакетов от
юзеров
U1=exprnd(MeanUserToBr(1), RpMax, 1);
U2=exprnd(MeanUserToBr(2), RpMax, 1);
U3=exprnd(MeanUserToBr(3), RpMax, 1);
U4=exprnd(MeanUserToBr(4), RpMax, 1);
MeanExpon=[mean(U1) mean(U2) mean(U3) mean(U4)];
StDevExpon=[std(U1) std(U2) std(U3) std(U4)];
MinExpon=[min(U1) min(U2) min(U3) min(U4)];
MaxExpon=[max(U1) max(U2) max(U3) max(U4)];

% %-----
1 U1(1,2)=t0+U1(1,1); % время поступления первого пакета от юзера в брокер
2 U2(1,2)=t0+U2(1,1); % время поступления первого пакета от юзера в брокер
3 U3(1,2)=t0+U3(1,1); % время поступления первого пакета от юзера в брокер
4 U4(1,2)=t0+U4(1,1); % время поступления первого пакета от юзера в брокер

for row=2:1:RpMax
    U1(row,2)=U1(row-1,2)+U1(row,1); % время поступления пакетов от юзеров в
брокер1 %% t(from)
    U2(row,2)=U2(row-1,2)+U2(row,1); % в брокер 2
    U3(row,2)=U3(row-1,2)+U3(row,1); % в брокер 3
    U4(row,2)=U4(row-1,2)+U4(row,1); % в брокер 4
end

W =[90 76 30 19 ];

%% генерация пакетов от каждого брокера к другими брокерам
MeanBrToBr= [ 0    156   82   59;
              156   0    130  107;
              82   130   0    118;
              59   107  118   0 ];

%v2 для равномерного закона
MinBrToBr= [ 0    142   68   45;
            142   0    116   93;
            68   116   0    104;
            45   93   104   0];
MaxBrToBr= [0    170   96   73;
            170   0    144  121;
            96   144   0    132;
            73   121  132   0];

```

```

    U1(1:RpMax,3)=(MaxBrToBr(1,2)-MinBrToBr(1,2)).*rand(RpMax,1) +
    MinBrToBr(1,2); %% интервалы между грабов (RP) от брокера 1 в брокер 2
    U1(1:RpMax,5)=(MaxBrToBr(1,3)-MinBrToBr(1,3)).*rand(RpMax,1) +
    MinBrToBr(1,3); %% интервалы между грабов (RP) от брокера 1 в брокер 3
    U1(1:RpMax,7)=(MaxBrToBr(1,4)-MinBrToBr(1,4)).*rand(RpMax,1) +
    MinBrToBr(1,4); %% интервалы между грабов (RP) от брокера 1 в брокер 4

    U2(1:RpMax,3)=(MaxBrToBr(2,1)-MinBrToBr(2,1)).*rand(RpMax,1) +
    MinBrToBr(2,1); %% интервалы между грабов (RP) от брокера 2 в брокер 1
    U2(1:RpMax,5)=(MaxBrToBr(2,3)-MinBrToBr(2,3)).*rand(RpMax,1) +
    MinBrToBr(2,3); %% интервалы между грабов (RP) от брокера 2 в брокер 3
    U2(1:RpMax,7)=(MaxBrToBr(2,4)-MinBrToBr(2,4)).*rand(RpMax,1) +
    MinBrToBr(2,4); %% интервалы между грабов (RP) от брокера 2 в брокер 4

    U3(1:RpMax,3)=(MaxBrToBr(3,1)-MinBrToBr(3,1)).*rand(RpMax,1) +
    MinBrToBr(3,1); %% интервалы между грабов (RP) от брокера 3 в брокер 1
    U3(1:RpMax,5)=(MaxBrToBr(3,2)-MinBrToBr(3,2)).*rand(RpMax,1) +
    MinBrToBr(3,2); %% интервалы между грабов (RP) от брокера 3 в брокер 2
    U3(1:RpMax,7)=(MaxBrToBr(3,4)-MinBrToBr(3,4)).*rand(RpMax,1) +
    MinBrToBr(3,4); %% интервалы между грабов (RP) от брокера 3 в брокер 4

    U4(1:RpMax,3)=(MaxBrToBr(4,1)-MinBrToBr(4,1)).*rand(RpMax,1) +
    MinBrToBr(4,1); %% интервалы между грабов (RP) от брокера 4 в брокер 1
    U4(1:RpMax,5)=(MaxBrToBr(4,2)-MinBrToBr(4,2)).*rand(RpMax,1) +
    MinBrToBr(4,2); %% интервалы между грабов (RP) от брокера 4 в брокер 2
    U4(1:RpMax,7)=(MaxBrToBr(4,3)-MinBrToBr(4,3)).*rand(RpMax,1) +
    MinBrToBr(4,3); %% интервалы между грабов (RP) от брокера 4 в брокер 3

    for row=1:1:RpMax %% t(to)
        U1(row,4)=U1(row,2)+U1(row,3); % от брокера 1 в брокер 2 %% t(to)
        U1(row,6)=U1(row,2)+U1(row,5); % от брокера 1 в брокер 3
        U1(row,8)=U1(row,2)+U1(row,7); % от брокера 1 в брокер 4
        %U1(row,9)= fix(U2(:,2)/I0)+1; % от брокера 1 [Ix-2]from

        U2(row,4)=U2(row,2)+U2(row,3); % от брокера 2 в брокер 1
        U2(row,6)=U2(row,2)+U2(row,5); % от брокера 2 в брокер 3
        U2(row,8)=U2(row,2)+U2(row,7); % от брокера 2 в брокер 4
        %U2(row,9)= fix(U2(:,2)/I0)+1; % от брокера 2 [Ix-2]from

        U3(row,4)=U3(row,2)+U3(row,3); % от брокера 3 в брокер 1
        U3(row,6)=U3(row,2)+U3(row,5); % от брокера 3 в брокер 2
        U3(row,8)=U3(row,2)+U3(row,7); % от брокера 3 в брокер 4
        %U3(row,9)= fix(U2(:,2)/I0)+1; % от брокера 3 [Ix-2]from

        U4(row,4)=U4(row,2)+U4(row,3); % от брокера 4 в брокер 1
        U4(row,6)=U4(row,2)+U4(row,5); % от брокера 4 в брокер 2
        U4(row,8)=U4(row,2)+U4(row,7); % от брокера 4 в брокер 3
        %U4(row,9)= fix(U2(:,2)/I0)+1; % от брокера 4 [Ix-2]from
    end

    SKOcheck =[ 0                std(U1(:,3))      std(U1(:,5))      std(U1(:,7))
                std(U2(:,3)) 0                std(U2(:,5))      std(U2(:,7))
                std(U3(:,3)) std(U3(:,5))      0                std(U3(:,7))
                std(U4(:,3)) std(U4(:,5))      std(U4(:,7))      0];
    SKOcheckMax =max(SKOcheck()) ;
    % fix - целая часть числа
    I0=295;

    U1(:,9)= fix(U1(:,2)/I0)+1; % номер интервала выхода от брокера 1
    U2(:,9)= fix(U2(:,2)/I0)+1; % номер интервала выхода пакета от брокера 2
    U3(:,9)= fix(U3(:,2)/I0)+1; % номер интервала выхода пакета от брокера 3
    U4(:,9)= fix(U4(:,2)/I0)+1; % номер интервала выхода пакета от брокера 4

```

```

U1(:,10)=1; % Br(from)
U2(:,10)=2; % Br(from)
U3(:,10)=3; % Br(from)
U4(:,10)=4; % Br(from)

U_All=[U1; U2; U3; U4];

Br1All0=[U1(:,2) U1(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) fix(U1(:,2)/I0)+1 ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U2(:,4) U2(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) fix(U2(:,4)/I0)+1 ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U3(:,4) U3(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) fix(U3(:,4)/I0)+1 ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U4(:,4) U4(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) fix(U4(:,4)/I0)+1 ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)] ;
Br1All0((RpMax+1):(2*RpMax), 3)=2; % пакеты от брокера 2
Br1All0((2*RpMax+1):(3*RpMax), 3)=3; % пакеты от брокера 3
Br1All0((3*RpMax+1):(4*RpMax), 3)=4; % пакеты от брокера 4

Br2All0=[U2(:,2) U2(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U2(:,2)/I0)+1 2*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U1(:,4) U1(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U1(:,4)/I0)+1 2*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U3(:,6) U3(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U3(:,6)/I0)+1 2*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U4(:,6) U4(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U4(:,6)/I0)+1 2*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)];
Br2All0(1:RpMax, 3)=2; % пакеты от брокера 2
Br2All0((2*RpMax+1):(3*RpMax), 3)=3; % пакеты от брокера 3
Br2All0((3*RpMax+1):(4*RpMax), 3)=4; % пакеты от брокера 4

Br3All0=[U3(:,2) U3(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U3(:,2)/I0)+1 3*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U1(:,6) U1(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U1(:,6)/I0)+1 3*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U2(:,6) U2(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U2(:,6)/I0)+1 3*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U4(:,8) U4(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U4(:,8)/I0)+1 3*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)];
Br3All0(1:RpMax, 3)=3; % пакеты от брокера 3
Br3All0((2*RpMax+1):(3*RpMax), 3)=2; % пакеты от брокера 2
Br3All0((3*RpMax+1):(4*RpMax), 3)=4; % пакеты от брокера 4
Br4All0=[U4(:,2) U4(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U4(:,2)/I0)+1 4*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U1(:,8) U1(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U1(:,8)/I0)+1 4*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);
    U2(:,8) U2(:,9) ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U2(:,8)/I0)+1 4*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1);

```

```

        U3(:,8)   U3(:,9)   ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)
-ones(RpMax,1) -ones(RpMax,1) fix(U3(:,8)/I0)+1 4*ones(RpMax,1) -
ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1) -ones(RpMax,1)];
    Br4All0(1:RpMax, 3)=4; % пакеты от брокера 4
    Br4All0((2*RpMax+1):(3*RpMax), 3)=2; % пакеты от брокера 2
    Br4All0((3*RpMax+1):(4*RpMax), 3)=3; % пакеты от брокера 3

    BrAll0= [Br1All0;Br2All0;Br3All0;Br4All0];
    BrAll = sortrows(BrAll0); % сортировка по времени для всех брокеров
for j=1:1:length(BrAll)
    if ((BrAll(j,1)-(BrAll(j,8)-1)*I0)<W(1,BrAll(j,9)))
        BrAll(j,10)=1; % tempWx(to)
    else
        BrAll(j,11)=1; % tempIx(to)
    end
end
end
Ixmax=max(BrAll(:,2));
Nst=zeros(Ixmax+2,4);
Nst(1,:)=[10 10 10 10]; % стартовый номер пакетов
Nst(2,:)=[10 10 10 10]; % стартовый номер пакетов
for Brx=1:1:max(BRnum)
    for Ixx=1:1:(Ixmax) % -2
        AtempW= find( (BrAll(:,9)==Brx) & (BrAll(:,8)==Ixx) & (BrAll(:,10)
==1) );
        tempW=1;
        for y=1:1:length(AtempW)
            BrAll(AtempW(y),10)=tempW;
            tempW=tempW+1;
        end
        AtempI= find( (BrAll(:,9)==Brx) & (BrAll(:,8)==Ixx) & (BrAll(:,11)
==1) );
        Next=1;
        for y=1:1:length(AtempI)
            BrAll(AtempI(y),12)=Next;
            Next=Next+1;
        end
    end
end
end

Lat1=zeros(1,5);
WrongLine1=[0];
for j=1:1:length(BrAll)
    if (BrAll(j,3)==BrAll(j,9)) %пакет от юзера
        Br_t_to= find((BrAll(:, 13)==BrAll(j,13))&(BrAll(:, 9)==1));
        if (BrAll(j,3)==1) % поступил в Br1
            Br_t_to=j;
            if(size(Br_t_to,1)~=1)
                WrongLine1=[WrongLine1; j];
            end
        end
        NewRP=[BrAll(j,1) BrAll(j,9) BrAll(j,13) BrAll(Br_t_to,1)
BrAll(Br_t_to,15)];
        if (size(NewRP,2)~=5)
            NewRP=[BrAll(j,1) BrAll(j,13) BrAll(j,9) 0 0 ];
            WrongLine1=[WrongLine1; j];
        end
        Lat1=[Lat1; NewRP;];
    end
end
end

% от всех юзеров к Br2
Lat2=zeros(1,5);
WrongLine2=[0];

```

```

for j=1:1:length(BrAll)
    if (BrAll(j,3)==BrAll(j,9)) %пакет от юзера
        NewRP=[BrAll(j,1) BrAll(j,9) BrAll(j,13) BrAll(Br_t_to,1)
BrAll(Br_t_to,15)];
        Lat2=[Lat2; NewRP;];
    end
end

% от всех юзеров к Br3
Lat3=zeros(1,5);
WrongLine3=[0];
for j=1:1:length(BrAll)
    if (BrAll(j,3)==BrAll(j,9)) %пакет от юзера
        NewRP=[BrAll(j,1) BrAll(j,9) BrAll(j,13) BrAll(Br_t_to,1)
BrAll(Br_t_to,15)];
        if (size(NewRP,2)~=5)
            NewRP=[BrAll(j,1) BrAll(j,13) BrAll(j,9) 0 0 ];
            WrongLine3=[WrongLine3; j];
        end
        Lat3=[Lat3; NewRP;];
    end
end

% от всех юзеров к Br4
Lat4=zeros(1,5);
WrongLine4=[0];
for j=1:1:length(BrAll)
    if (BrAll(j,3)==BrAll(j,9)) %пакет от юзера
        Lat4=[Lat4; NewRP;];
    end
end

% задержка формирования Ng1
Latens5_4expon=[0 0 0 0];
for jj=2:1:length(Lat1)
    if Lat1(jj,5)~=0
        14_1=Lat1(jj,5)-Lat1(jj,4);
    else
        14_1=0;
    end
    if Lat2(jj,5)~=0
        14_2=Lat2(jj,5)-Lat2(jj,4);
    else
        14_2=0;
    end
    if Lat3(jj,5)~=0
        14_3=Lat3(jj,5)-Lat3(jj,4);
    else
        14_3=0;
    end
    if Lat4(jj,5)~=0
        14_4=Lat4(jj,5)-Lat4(jj,4);
    else
        14_4=0;
    end
    Latens5_4expon=[Latens5_4expon; 14_1 14_2 14_3 14_4; ];
end

Latens5_4Pareto=[0 0 0 0];
for jj=2:1:length(Lat1Pareto)
    if Lat1Pareto(jj,5)~=0
        14_1Pareto=Lat1Pareto(jj,5)-Lat1Pareto(jj,4);
    else
        14_1Pareto=0;
    end
end

```

```

end
if Lat2Pareto(jj,5)~=0
    l4_2Pareto=Lat2Pareto(jj,5)-Lat2Pareto(jj,4);
else
    l4_2Pareto=0;
end
if Lat3Pareto(jj,5)~=0
    l4_3Pareto=Lat3Pareto(jj,5)-Lat3Pareto(jj,4);
else
    l4_3Pareto=0;
end
if Lat4Pareto(jj,5)~=0
    l4_4Pareto=Lat4Pareto(jj,5)-Lat4Pareto(jj,4);
else
    l4_4Pareto=0;
end
Latens5_4Pareto=[Latens5_4Pareto; l4_1Pareto l4_2Pareto l4_3Pareto
l4_4Pareto; ];
end

%-----
figure(4);
subplot(2,2, 1);
ecdf(Latens5_4expon(:,1));
hold on
ecdf(Latens5_4unif(:,1));
ecdf(Latens5_4Pareto(:,1));
hold off

subplot(2,2, 2);
ecdf(Latens5_4expon(:,2));
hold on
ecdf(Latens5_4unif(:,2));
ecdf(Latens5_4Pareto(:,2));
hold off

subplot(2,2, 3);
ecdf(Latens5_4expon(:,3));
hold on
ecdf(Latens5_4unif(:,3));
ecdf(Latens5_4Pareto(:,3));
hold off

subplot(2,2, 4);
ecdf(Latens5_4expon(:,4));
hold on
ecdf(Latens5_4unif(:,4));
ecdf(Latens5_4Pareto(:,4));
hold off

```