Comparing concertina and nomao pagination methodology

Table of Contents

we here use two different product lists to assess the performance of the 2 methodologies : small size : humour et jeux/l-10535 big size : batteries-chargeurs/l-1070929

Parameters for the first middle size product list

humour et jeux/l-10535

```
nb_product=436.*23;
% with 23 products per page
% nb_page=436;
nb_page=436; %#ok<NASGU>
% the depth of click we want to explore
depth_click=10;
% total number of pages to display
Ntotal=436;
% total number of displayable references in pagination
Nlinks=15;
% notice that relevant cases are Ntotal >> Nlink
% padding parameter
Npadding=3;
% our pagination distribution computation
concertina_distrib=@(x)concertina_compute_pagination(x, Nlinks, Npadding, Ntotal);
nomao_distrib=@(x)nomao_compute_pagination(x, Nlinks, Ntotal);
```

Computing concertina reached pages per number of clicks

we compute the initial distribution

```
distribution=concertina_distrib(1);
click=1;
my_evolving_distrib=cell(depth_click,1);
```

```
while (click <= depth_click)
    %  links_to_add instantiation
    new_list=[];
    for link=1:size(distribution,1)
        % we here compute the number of pages reachable at the variable click
        new_list = union(new_list,concertina_distrib(distribution(link)));
    end
    my_evolving_distrib{click}=new_list;
    distribution=new_list;
    %  links_to_add=added_links+newlist
    % we here list the number of pages reached at click number
    click=click+1;
end
concertina_length_vector = cellfun(@length, my_evolving_distrib);
```

Computing concertina reached pages per number of clicks

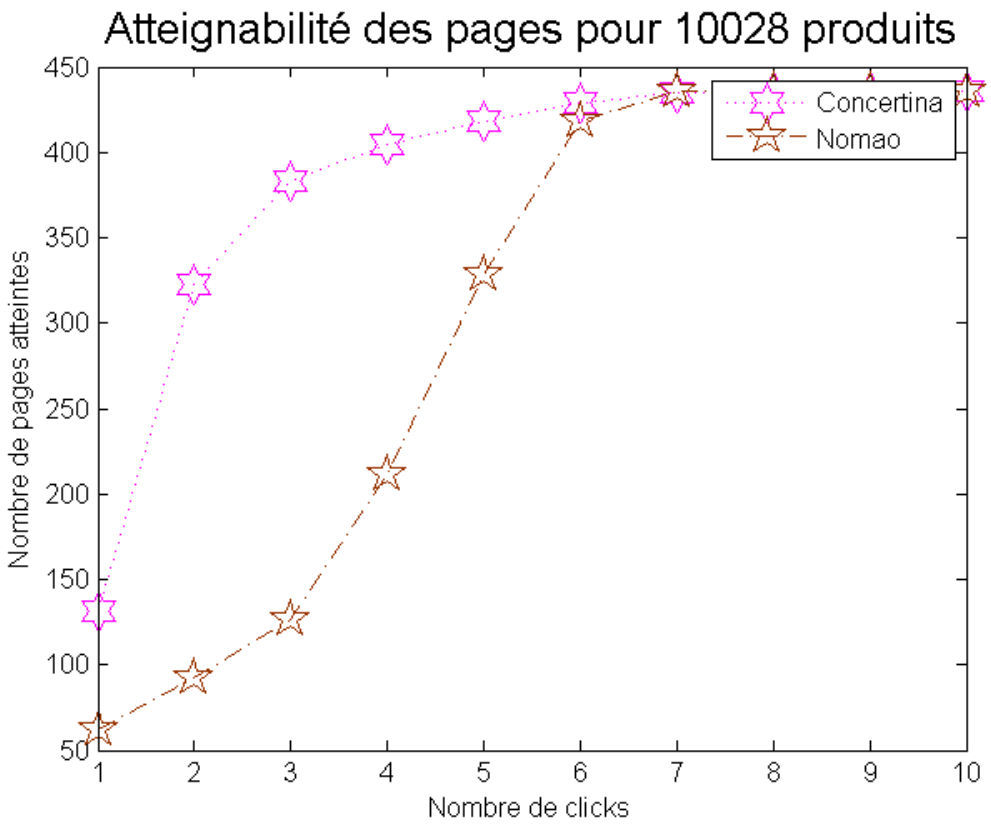
we compute the initial distribution

```
distribution=nomao_distrib(1);
click=1;
my_evolving_distrib=cell(depth_click,1);
while (click <= depth_click)
    %  links_to_add instantiation
    new_list=[];
    for link=1:size(distribution,1)
        % we here compute the number of pages reachable at the variable click
        new_list = union(new_list,nomao_distrib(distribution(link)));
    end
    my_evolving_distrib{click}=new_list;
    distribution=new_list;
    %  links_to_add=added_links+newlist
    % we here list the number of pages reached at click number
    click=click+1;
end
nomao_distrib_length_vector = cellfun(@length, my_evolving_distrib);
```

Visualizing the evolution

making a nice plot

```
createfigure([concertina_length_vector,nomao_distrib_length_vector],['Atteignabili
```



Parameters for the second giant size product list

batterie et chargeur batteries-chargeurs/l-1070929

```
nb_product=600297;
% with 23 products per page
% nb_page=26304;
nb_page=600297./23;
% the depth of click we want to explore
depth_click=200;
% total number of pages to display
Ntotal=26304;
% total number of displayable references in pagination
Nlinks=14;
% notice that relevant cases are Ntotal >> Nlink
% padding parameter
Npadding=3;
% our pagination distribution computation
% our pagination distribution computation
concertina_distrib=@(x)concertina_compute_pagination(x, Nlinks, Npadding, Ntotal);
nomao_distrib=@(x)nomao_compute_pagination(x, Nlinks, Ntotal);
```

Computing concertina reached pages per number of clicks

we compute the initial distribution

```
distribution=concertina_distrib(1);
click=1;
my_evolving_distrib=cell(depth_click,1);
while (click <= depth_click)
    %  links_to_add instantiation
    new_list=[];
    for link=1:size(distribution,1)
        % we here compute the number of pages reachable at the variable click
        new_list = union(new_list,concertina_distrib(distribution(link)));
    end
    my_evolving_distrib{click}=new_list;
    distribution=new_list;
    %  links_to_add=added_links+newlist
    % we here list the number of pages reached at click number
    click=click+1;
end
concertina_length_vector = cellfun(@length, my_evolving_distrib);
```

Computing concertina reached pages per number of clicks

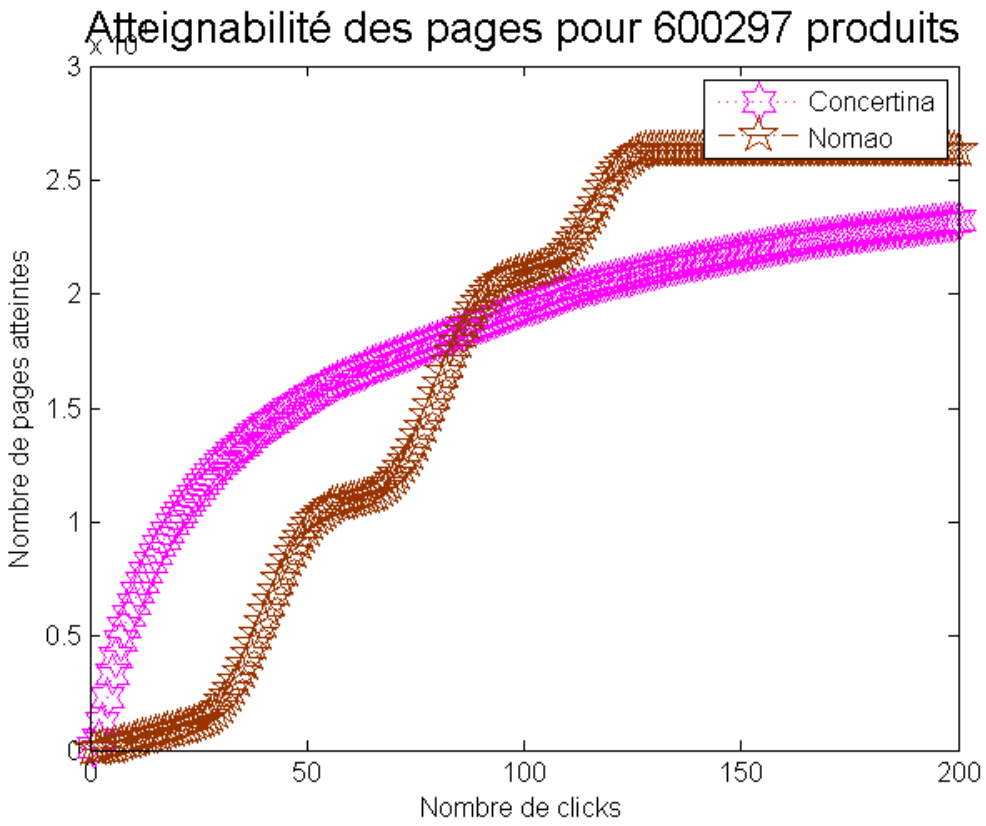
we compute the initial distribution

```
distribution=nomao_distrib(1);
click=1;
my_evolving_distrib=cell(depth_click,1);
while (click <= depth_click)
    %  links_to_add instantiation
    new_list=[];
    for link=1:size(distribution,1)
        % we here compute the number of pages reachable at the variable click
        new_list = union(new_list,nomao_distrib(distribution(link)));
    end
    my_evolving_distrib{click}=new_list;
    distribution=new_list;
    %  links_to_add=added_links+newlist
    % we here list the number of pages reached at click number
    click=click+1;
end
nomao_distrib_length_vector = cellfun(@length, my_evolving_distrib);
```

Visualizing the evolution

making a nice plot

```
createfigure([concertina_length_vector,nomao_distrib_length_vector],['Atteignabili
```



Published with MATLAB® R2014a