

Bibliografía

Amaldi, E., Capone, A., Cesana, M., Malucelli, F. and Palazzo, F. (2004), *WLAN Coverage Planning: Optimization Models and Algorithms*, in Proceedings of the IEEE Vehicular Technology Conference (VTC 2004-Spring), May 17-19, Milan, Italy.

Amaldi, E., Capone, A., Cesana, M., Fratta, L. and Malucelli, F. (2005), *Algorithms for WLAN Coverage Planning*. Mobile and Wireless Systems, in Lecture Notes in Computer Science 3427, pp. 52-65.

Bahri, A. and Chamberland, S. (2005), *On the Wireless Local Area Network Design Problem with Performance Guarantees*, Computers Networks 48, 856-866.

Cardei, M. and Wu, J. (2006), *Energy-Efficient Coverage Problems in Wireless Ad-Hoc Sensor Networks*, Computer Communications 29, 413-420.

Garey, M.R. and Johnson, D.S. (1979), *Computers and Intractability. A guide to the Theory of NP-Completeness*, W.H. Freeman and Company, New York.

Huang, C-F. and Tseng, Y-C. (2003), *The Coverage Problem in a Wireless Sensor Network*, in WSNA'03: Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, (ACM Press 2003) pp. 115-121.

Liu, B. and Towsley, D. (2004), *A Study of the Coverage of Large-scale Sensor Networks*, in the First IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS 2004), pp. 475-483.

Meester, R. and Roy, R. (1996), *Continuum Percolation*, Cambridge University Press.

Meguerdichian, S., Koushanfar, F., Potkonjak, M. and Srivastava, M.B. (2001), *Coverage Problems in Wireless Ad-Hoc Sensor Networks*, Proceedings of the IEEE Infocom'01, pp. 1380-1387.

Megerian, S., Koushanfar, F., Potkonjak, M. and Srivastava, M.B. (2005), *Worst and Best-Case Coverage in Sensor Networks*, IEEE Transactions on Mobile Computing 4, 84-92.

Tian, D. and Georganas, N.D. (2002), *A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks*, in the First ACM International Workshop on Wireless Sensor Networks and Applications, pp. 32-41.

Slijepcevic, S. and Potkonjak, M. (2001), *Power Efficient Organization of Wireless Sensor Networks*, in Proceedings of IEEE international Conference on Communications (ICC'01), pp. 472-476.

Yang, S., Dai, F., Cardei, M. and Wu, J. (2005), *On Multiple Point Coverage in Wireless Sensor Networks*, in Proceedings of the 2nd. IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'05).

Ye, F., Zhong, G., Lu, S. and Zhang, L. (2003), *Peas: A Robust Energy Conserving Protocol for Long-Lived Sensor Networks*, in Proceedings of the 23rd IEEE International Conference on Distributed Computing Systems.

ANEXO

Códigos MATLAB


```
function y=matriz_n_m(n,m)

S=round(rand(n,m));

C=zeros(n,m);
indice_c_fila=1;
indice_c_columna=1;
C_no_usados=zeros(1,n);
columnas_ceros=zeros(1,m);
indice_cero=1;
Sprov=S;
no_cumple=0;
contador_unos=0;
sensores_malos=zeros(1,n);

for j=1:m
    contador_unos=0;
    for i=1:n
        contador_unos=contador_unos+Sprov(i,j);
    end;
    if contador_unos==0
        no_cumple=1;
    end;
end;

while no_cumple==0

    vector_ceros=0;
    iter_primera=0;
    sensores_malos=zeros(1,n);
    ind_malo=1;

    while vector_ceros==0
        sensor_malo=0;

        obj_elegido=0;
        suma_unos_min=n;

        for j=1:m
            suma_unos=0;
            for i=1:n
                suma_unos=suma_unos+Sprov(i,j);
            end;

            if suma_unos<suma_unos_min
                obj_elegido=j;
                suma_unos_min=suma_unos;
            end;
        end;
    end;
end;
```

```
end;
sensor_elegido=0;
suma_unos_max=0;

for i=1:n
    suma_unos=0;
    for j=1:m
        suma_unos=suma_unos+Sprov(i,j);
    end;
    if iter_primera==0
        if (suma_unos>suma_unos_max & Sprov(i,obj_elegido)==1)
            sensor_elegido=i;
            suma_unos_max=suma_unos;
        end;
    else
        for h=1:length(sensores_malos)
            if (suma_unos>suma_unos_max & i~=sensores_malos(h))
                sensor_elegido=i;
                suma_unos_max=suma_unos;
            end;
        end;
    end;
    contador_columnas_ceros=0;
    contador_SProv_ceros=0;
    for k=1:length(columnas_ceros)
        if (columnas_ceros(k)~=0 & sensor_elegido~=0 & iter_primera==1)
            contador_columnas_ceros=contador_columnas_ceros+1;
        if (Sprov(sensor_elegido,columnas_ceros(k))==0 & iter_primera==1)
            contador_SProv_ceros=contador_SProv_ceros+1;
        end
    end;
    if (contador_SProv_ceros<contador_columnas_ceros & iter_primera==1)
        sensor_malo=0;
        enc=1;
    else
        if (contador_SProv_ceros>=contador_columnas_ceros & iter_primera==1)
            sensor_malo=1;
        end;
    end;
end;

if sensor_malo==0

    C(indice_c_fila,indice_c_columna)=sensor_elegido;
    indice_c_columna=indice_c_columna+1;
```

```
for j=1:m
    if (Sprov(sensor_elegido,j)==0 & iter_primera==0)
        columnas_ceros(indice_cero)=j;
        indice_cero=indice_cero+1;
    end;
end;

if iter_primera==0
    hay_distinto_cero=0;
    for h=1:m
        if columnas_ceros(h)~=0
            hay_distinto_cero=1;
        end;
    end;
    if hay_distinto_cero==0
        vector_ceros=1;
    end;
end;

if iter_primera==1
    for k=1:length(columnas_ceros)
        if columnas_ceros(k)~=0
            if Sprov(sensor_elegido,columnas_ceros(k))==1
                columnas_ceros(k)=0;
            end;
        end;
    end;
end;

for i=1:n
    if i==sensor_elegido
        Sprov(i,:)=zeros(1,m);
    end;
end;

no_cumple=0;
contador_unos=0;

for j=1:m
    contador_unos=0;
    for i=1:n
        contador_unos=contador_unos+Sprov(i,j);
    end;
    if contador_unos==0
        no_cumple=1;
    end;
end;
```

```
        if iter_primera==1
            cont_cer=0;
            for k=1:length(columnas_ceros)
                if columnas_ceros(k)~=0
                    cont_cer=cont_cer+1;
                end;
            end;
            if cont_cer==0
                vector_ceros=1;
            end;
        end;

    else
        sensores_malos(ind_malo)=sensor_elegido;
        ind_malo=ind_malo+1;
    end;

    iter_primera=1;

end;

indice_c_columna=1;
indice_c_fila=indice_c_fila+1;

end;

cont_c_no_usados=1;
hay_no_usados=0;

for i=1:n
    valor_hallado=0;
    for k1=1:n
        for k2=1:m
            if C(k1,k2)==i
                valor_hallado=1;
            end;
        end;
    end;
    if valor_hallado==0
        C_no_usados(cont_c_no_usados)=i;
        cont_c_no_usados=cont_c_no_usados+1;
        hay_no_usados=1;
    end;
end;

fprintf('Los subconjuntos de sensores C son:\n');
```



```
for i=1:n
    if C(i,1)~=0
        valor_cero=0;
        fprintf('C%d={',i);
        for j=1:m
            if (C(i,j)~=0 & valor_cero==0)
                fprintf('%d',C(i,j));
                if C(i,j+1)~=0
                    fprintf(',');
                end;
            else
                valor_cero=1;
            end;
        end;
        fprintf('}\n');
    end;
end;

if hay_no_usados==1
    fprintf('Los sensores no usados son:\n{');
    for k=1:n
        if C_no_usados(k)~=0
            fprintf('%d',C_no_usados(k));
            if C_no_usados(k+1)~=0
                fprintf(',');
            end;
        end;
    end;
    fprintf('}');
end;
```

```
function y=modificacion(n,m)

S=round(rand(n,m));

C=zeros(n,m);
indice_c_fila=1;
indice_c_columna=1;
C_no_usados=zeros(1,n);
columnas_ceros=zeros(1,m);
indice_cero=1;
Sprov=S;
no_cumple=0;
contador_unos=0;
sensores_malos=zeros(1,n);

for j=1:m
    contador_unos=0;
    for i=1:n
        contador_unos=contador_unos+Sprov(i,j);
    end;
    if contador_unos==0
        no_cumple=1;
    end;
end;

while no_cumple==0

    vector_ceros=0;
    iter_primera=0;
    sensores_malos=zeros(1,n);
    ind_malo=1;

    while vector_ceros==0
        sensor_malo=0;

        obj_elegido=0;
        suma_unos_min=n;

        for j=1:m
            suma_unos=0;
            for i=1:n
                suma_unos=suma_unos+Sprov(i,j);
            end;

            if suma_unos<suma_unos_min
                obj_elegido=j;
                suma_unos_min=suma_unos;
            end;
        end;
    end;
end;
```

```
end;

sensor_elegido=0;
suma_unos_max=0;
bonus_max=0;

for i=1:n
    suma_unos=0;
    for j=1:m
        suma_unos=suma_unos+Sprov(i,j);
    end;
    if iter_primera==0
        if (suma_unos>suma_unos_max & Sprov(i,obj_elegido)==1)
            sensor_elegido=i;
            suma_unos_max=suma_unos;
        end;
    else
        for h=1:length(sensores_malos)
            if (suma_unos>suma_unos_max & i~=sensores_malos(h))
                sensor_elegido=i;
                suma_unos_max=suma_unos;
            end;
        end;
    end;
    contador_columnas_ceros=0;
    contador_SProv_ceros=0;

    bonus=0;
    for k=1:length(columnas_ceros)
        if (columnas_ceros(k)~=0 & sensor_elegido~=0 & iter_primera==1)
            contador_columnas_ceros=contador_columnas_ceros+1;
        if (Sprov(sensor_elegido,columnas_ceros(k))==0 & iter_primera==1)
            contador_SProv_ceros=contador_SProv_ceros+1;
        else
            bonus=bonus+1;
        end;
    end;
end;

if (contador_SProv_ceros<contador_columnas_ceros & iter_primera==1 &
bonus>bonus_max)
    sensor_malo=0;
    bonus_max=bonus;
    enc=1;
else
    if (contador_SProv_ceros>=contador_columnas_ceros & iter_primera==1)
        sensor_malo=1;
    end;
end;
```

```
        end;
    end;
    if sensor_malo==0
        C(indice_c_fila,indice_c_columna)=sensor_elegido;
        indice_c_columna=indice_c_columna+1;

        for j=1:m
            if (Sprov(sensor_elegido,j)==0 & iter_primera==0)
                columnas_ceros(indice_cero)=j;
                indice_cero=indice_cero+1;
            end;
        end;

        if iter_primera==0
            hay_distinto_cero=0;
            for h=1:m
                if columnas_ceros(h)~=0
                    hay_distinto_cero=1;
                end;
            end;
            if hay_distinto_cero==0
                vector_ceros=1;
            end;
        end;

        if iter_primera==1
            for k=1:length(columnas_ceros)
                if columnas_ceros(k)~=0
                    if Sprov(sensor_elegido,columnas_ceros(k))==1
                        columnas_ceros(k)=0;
                    end;
                end;
            end;
        end;

        for i=1:n
            if i==sensor_elegido
                Sprov(i,:)=zeros(1,m);
            end;
        end;

        no_cumple=0;
        contador_unos=0;

        for j=1:m
            contador_unos=0;
            for i=1:n
                contador_unos=contador_unos+Sprov(i,j);
            end;
        end;
    end;
end;
```

```
end;
if contador_unos==0
    no_cumple=1;
end;
end;

if iter_primera==1
    cont_cer=0;
    for k=1:length(columnas_ceros)
        if columnas_ceros(k)~=0
            cont_cer=cont_cer+1;
        end;
    end;
    if cont_cer==0
        vector_ceros=1;
    end;
end;

else
    sensores_malos(ind_malo)=sensor_elegido;
    ind_malo=ind_malo+1;
end;

iter_primera=1;

end;

indice_c_columna=1;
indice_c_fila=indice_c_fila+1;

end;

cont_c_no_usados=1;
hay_no_usados=0;

for i=1:n
    valor_hallado=0;
    for k1=1:n
        for k2=1:m
            if C(k1,k2)==i
                valor_hallado=1;
            end;
        end;
    end;
end;

if valor_hallado==0
    C_no_usados(cont_c_no_usados)=i;
    cont_c_no_usados=cont_c_no_usados+1;
end;
```

```
    hay_no_usados=1;
end;
end;

fprintf('Los subconjuntos de sensores C son:\n');

for i=1:n
    if C(i,1)~=0
        valor_cero=0;
        fprintf('C%d={',i);
        for j=1:m
            if (C(i,j)~=0 & valor_cero==0)
                fprintf('%d',C(i,j));
                if C(i,j+1)~=0
                    fprintf(',');
                end;
            else
                valor_cero=1;
            end;
        end;
        fprintf('}\n');
    end;
end;

if hay_no_usados==1
    fprintf('Los sensores no usados son:\n{');
    for k=1:n
        if C_no_usados(k)~=0
            fprintf('%d',C_no_usados(k));
            if C_no_usados(k+1)~=0
                fprintf(',');
            end;
        end;
    end;
    fprintf('}');
end;
```

```
function y=tiempo(n,m,t)

S=round(rand(n,m));
x=ceil(t*rand(1,n));
C=zeros(n,m);
indice_c_fila=1;
indice_c_columna=1;
C_no_usados=zeros(1,n);
columnas_ceros=zeros(1,m);
indice_cero=1;
Sprov=S;
no_cumple=0;
contador_unos=0;
sensores_malos=zeros(1,n);

for j=1:m
    contador_unos=0;
    for i=1:n
        contador_unos=contador_unos+Sprov(i,j);
    end;
    if contador_unos==0
        no_cumple=1;
    end;
end;

while no_cumple==0

    vector_ceros=0;
    iter_primera=0;
    sensores_malos=zeros(1,n);
    ind_malo=1;

    while vector_ceros==0
        sensor_malo=0;
        min_tiempo=t;

        obj_elegido=0;
        suma_unos_min=n;

        for j=1:m
            suma_unos=0;
            for i=1:n
                suma_unos=suma_unos+Sprov(i,j);
            end;

            if suma_unos<suma_unos_min
                obj_elegido=j;
                suma_unos_min=suma_unos;
            end;
        end;
    end;
end;
```

```
        end;
    end;
    sensor_elegido=0;
    suma_unos_max=0;

    for i=1:n
        suma_unos=0;
        for j=1:m
            suma_unos=suma_unos+Sprov(i,j);
        end;
        if iter_primera==0
            if (suma_unos>suma_unos_max & Sprov(i,obj_elegido)==1)
                sensor_elegido=i;
                suma_unos_max=suma_unos;
            end;
        else
            for h=1:length(sensores_malos)
                if (suma_unos>suma_unos_max & i~=sensores_malos(h) &
x(i)<min_tiempo)
                    sensor_elegido=i;
                    suma_unos_max=suma_unos;
                    min_tiempo=x(i);
                end;
            end;
        end;
        contador_columnas_ceros=0;
        contador_SProv_ceros=0;
        for k=1:length(columnas_ceros)
            if (columnas_ceros(k)~=0 & sensor_elegido~=0 & iter_primera==1)
                contador_columnas_ceros=contador_columnas_ceros+1;
            if (Sprov(sensor_elegido,columnas_ceros(k))==0 & iter_primera==1)
                contador_SProv_ceros=contador_SProv_ceros+1;
            end
        end;
        if (contador_SProv_ceros<contador_columnas_ceros & iter_primera==1)
            sensor_malo=0;
            enc=1;
        else
            if (contador_SProv_ceros>=contador_columnas_ceros & iter_primera==1)
                sensor_malo=1;
            end;
        end;
    end;

    if sensor_malo==0
        C(indice_c_fila,indice_c_columna)=sensor_elegido;
        indice_c_columna=indice_c_columna+1;
    end;
```



```
for j=1:m
    if (Sprov(sensor_elegido,j)==0 & iter_primera==0)
        columnas_ceros(indice_cero)=j;
        indice_cero=indice_cero+1;
    end;
end;

if iter_primera==0
    hay_distinto_cero=0;
    for h=1:m
        if columnas_ceros(h)~=0
            hay_distinto_cero=1;
        end;
    end;
    if hay_distinto_cero==0
        vector_ceros=1;
    end;
end;

if iter_primera==1
    for k=1:length(columnas_ceros)
        if columnas_ceros(k)~=0
            if Sprov(sensor_elegido,columnas_ceros(k))==1
                columnas_ceros(k)=0;
            end;
        end;
    end;
end;

for i=1:n
    if i==sensor_elegido
        Sprov(i,:)=zeros(1,m);
    end;
end;

no_cumple=0;
contador_unos=0;

for j=1:m
    contador_unos=0;
    for i=1:n
        contador_unos=contador_unos+Sprov(i,j);
    end;
    if contador_unos==0
        no_cumple=1;
    end;
end;
```

```
        if iter_primera==1
            cont_cer=0;
            for k=1:length(columnas_ceros)
                if columnas_ceros(k)~=0
                    cont_cer=cont_cer+1;
                end;
            end;
            if cont_cer==0
                vector_ceros=1;
            end;
        end;

    else
        sensores_malos(ind_malo)=sensor_elegido;
        ind_malo=ind_malo+1;
    end;

    iter_primera=1;
end;

indice_c_columna=1;
indice_c_fila=indice_c_fila+1;
end;

cont_c_no_usados=1;
hay_no_usados=0;

for i=1:n
    valor_hallado=0;
    for k1=1:n
        for k2=1:m
            if C(k1,k2)==i
                valor_hallado=1;
            end;
        end;
    end;
    if valor_hallado==0
        C_no_usados(cont_c_no_usados)=i;
        cont_c_no_usados=cont_c_no_usados+1;
        hay_no_usados=1;
    end;
end;

fprintf('Los subconjuntos de sensores C son:\n');

for i=1:n
    if C(i,1)~=0
        valor_cero=0;
```

```
fprintf('C%d={',i);
for j=1:m
    if (C(i,j)~=0 & valor_cero==0)
        fprintf('%d',C(i,j));
        if C(i,j+1)~=0
            fprintf(',');
        end;
    else
        valor_cero=1;
    end;
end;
fprintf('}\n');
end;
end;

if hay_no_usados==1
fprintf('Los sensores no usados son:\n{');
for k=1:n
    if C_no_usados(k)~=0
        fprintf('%d',C_no_usados(k));
        if C_no_usados(k+1)~=0
            fprintf(',');
        end;
    end;
end;
fprintf('}');

end;
```

