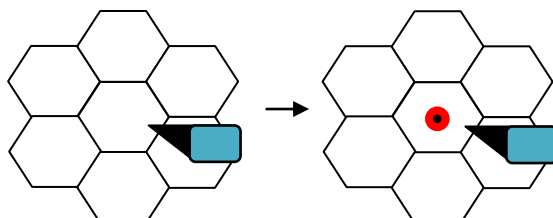


AERIUS CALCULATOR

Technische beschrijving geografische gridfuncties



datum 19 mei 2014

status Concept

versie 1.0

auteurs Bert Scholten, Maarten Schilpzand,
Diederik Metz



Colofon

Projectnaam	AERIUS
Locatie	Utrecht
Projectleider	Nico-Dirk van Loo
Contactpersoon	Nico-Dirk van Loo, 0646768026
Bijlage(n)	-
Auteurs	Bert Scholten, Maarten Schilpzand, Diederik Metz

Versiebeheer		
Versie	Datum	Korte beschrijving aanpassing
1.0	19-05-2014	Integraal document

Inhoud

INLEIDING	6
1 STANDAARDWAARDEN	7
2 AE_DETERMINE_RECEPTOR_ID_FROM_COORDINATES.....	8
3 AE_DETERMINE_RECEPTOR_IDS_IN_RECTANGLE.....	9
4 AE_DETERMINE_RECEPTOR_IDS_FROM_RECEPTOR_WITH_RADIUS	10
5 AE_IS_RECEPTOR_ID_AVAILABLE_ON_ZOOMLEVEL.....	12
6 AE_DETERMINE_COORDINATES_FROM_RECEPTOR_ID.....	13
7 AE_DETERMINE_RECEPTOR_IDS_FROM_RECEPTOR_BETWEEN_RADII.....	14
8 AE_DETERMINE_RECEPTOR_IDS_FROM_RECEPTOR_BETWEEN_RADII (ONLY WITHIN N2000 AREAS OR NOT).....	15

Inleiding

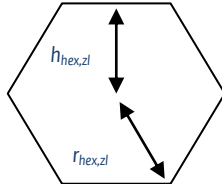
Wanneer een gebruiker een bron invoert in AERIUS zijn de x,y coördinaten van de bronlocatie bekend. Een geografische gridfunctie selecteert, op basis van deze x,y coördinaten, de rekenpunten die relevant zijn voor de berekening van de depositiebijdrage van de ingevoerde bron. Deze rekenpunten liggen in het midden van hexagonalen, die als vast raster over Nederland zijn neergelegd.

Elke rekenpunt heeft een uniek nummer (ID) dat overeenkomt met het ID van het bijbehorende hexagoon. Op basis hiervan kan een koppeling worden gemaakt met de geografische gegevens in de database. Per hexagoon zijn in de database gegevens opgenomen over bijvoorbeeld de Kritische Depositiewaarde (KDW), de achtergrondconcentratie en het habitatype.

Dit rapport beschrijft de verschillende geografische functies die in AERIUS worden toepast.

1 Standaardwaarden

De variabelen in de verschillende geografische functies komen voor een groot deel overeen. Dit hoofdstuk beschrijft de variabelen die in meerdere functies terugkomen.



Waarden

$$x_{min,bb} = 3604$$

$$x_{max,bb} = 287959$$

$$y_{min,bb} = 296800$$

$$y_{max,bb} = 629300$$

$$r_{hex,zl} = 100 \sqrt{\frac{2}{3\sqrt{3}}} \cdot 2^{zl-1}$$

$$h_{hex,zl} = \sqrt{3} \cdot r_{hex,zl}$$

$$w_{bs} = 1000$$

$$h_{bs} = 1000$$

$$\#hex_{hor} = \text{ceil} \left(\frac{\text{ceil} \left((x_{max,bb} - x_{min,bb}) \cdot \frac{3}{2} r_{hex,1} \right) + 1}{2} \right)$$

$$rec_{id,max} = \text{ceil} \left(\frac{2(y_{max,bb} - y_{min,bb})}{h_{hex,zl}} \right) \cdot \#hex_{hor}$$

$$\#bs_{hor} = \text{ceil} \left(\frac{x_{max,bb} - x_{min,bb}}{w_{bs}} \right) + 1$$

Omschrijving

$x_{min,bb}$ = Minimale x coördinaat van de bounding box

$x_{max,bb}$ = Maximale x coördinaat van de bounding box

$y_{min,bb}$ = Minimale y coördinaat van de bounding box

$y_{max,bb}$ = Maximale y coördinaat van de bounding box

$r_{hex,zl}$ =

"Straal" (lengte van het middenpunt naar hoekpunt) van een hexagon gegeven een zoomlevel zl

$h_{hex,zl}$ = Lengte van het middenpunt naar de top van een hexagon gegeven een zoomlevel zl

w_{bs} = Breedte van een "background square"

h_{bs} = Hoogte van een "background square"

$\#hex_{hor}$ = Aantal hexagonalen naast elkaar

$rec_{id,max}$ = Maximum receptor id

$\#bs_{hor}$ = Aantal "background squares" naast elkaar

rec_{db} = Receptoren in de database

2 ae_determine_receptor_id_from_coordinates

Functie voor het bepalen van een receptor_ID bij gegeven x, y coördinaten en zoomlevel. De functie maakt gebruik van het feit dat bij een hexagonaal grid er twee rechthoekige grids over elkaar liggen. Met de berekening van c_1 en c_2 wordt bepaald in welk rechthoekig grid gekeken moet worden. Vervolgens wordt gebruik gemaakt van het vaste aantal hexagonen op een regel.

Invoer

$x_{input}, y_{input}, z_{l_{input}}$

Berekening

$$x_1 = x_{input} - x_{min}$$

$$a_1 = x_1 \bmod (r_{hex, z_{l_{input}}})$$

$$a_2 = x_1 \bmod (r_{hex, z_l}) - \frac{3}{2} r_{hex, z_{l_{input}}}$$

$$y_1 = y_{input} - y_{min}$$

$$b_1 = y_1 \bmod (h_{hex, z_{l_{input}}})$$

$$b_2 = y_1 \bmod (h_{hex, z_{l_{input}}}) - \frac{h_{hex, z_{l_{input}}}}{2}$$

$$c_1 = a_1^2 + b_1^2$$

$$c_2 = a_2^2 + b_2^2$$

$$rec_{id} = \begin{cases} \#hex_{hor} (2 \cdot y_2 \text{quot} (h_{hex, z_{l_{input}}})) + x_1 \cdot \text{quot} (3 \cdot r_{hex, z_{l_{input}}}) + 1 \rightarrow c_1 > c_2 \\ \#hex_{hor} (2 \cdot y_2 \text{quot} (h_{hex, z_{l_{input}}}) + 1) + x_1 \cdot \text{quot} (3 \cdot r_{hex, z_{l_{input}}}) + 1 \rightarrow c_2 > c_1 \end{cases}$$

Uitvoer

rec_{id}

3 ae_determine_receptor_ids_in_rectangle

Functie voor het bepalen van de receptor ID's bij een gegeven rechthoek (bijvoorbeeld een background). Het algoritme begint met het bepalen van:

- het aantal hexagonen in de even en oneven rijen voor en in de rechthoek
- het aantal rijen onder en in de rechthoek

Vervolgens wordt bepaald wat de receptor ID's zijn van de laatste receptoren in de even en oneven rij voor de rechthoek. Met dit gegeven en het aantal hexagonen in de rechthoek worden de receptor ID's van zowel de even als oneven rijen bepaald. De samenvoeging van deze receptoren geeft de output. Hierbij wordt ook nog rekening gehouden met hexagonen die wel binnen Nederland moeten vallen.

Invoer

$x_{min,input}, x_{max,input}, y_{min,input}, y_{max,input}$

Berekening

$$x_1 = \max(x_{min,input}, x_{min,bb})$$

$$x_2 = \min(x_{max,input}, x_{max,bb})$$

$$y_1 = \max(y_{min,input}, y_{min,bb})$$

$$y_2 = \min(y_{max,input}, y_{max,bb})$$

$$\#hex_{hor,for,odd} = \text{ceil}\left(\frac{x_1 - x_{min,input}}{3 \cdot r_{hex,1}}\right)$$

$$\#hex_{hor,in,odd} = \text{ceil}\left(\frac{x_2 - x_{min,input}}{3 \cdot r_{hex,1}}\right) - \#hex_{hor,for,odd}$$

$$\#hex_{hor,for,even} = \text{trunc}\left(\frac{x_1 - x_{min,input}}{3 \cdot r_{hex,1}} + \frac{1}{2}\right)$$

$$\#hex_{hor,in,even} = \text{trunc}\left(\frac{x_2 - x_{min,input}}{3 \cdot r_{hex,1}} + \frac{1}{2}\right) - \#hex_{hor,for,even}$$

$$\#hex_{ver,under,odd} = \text{ceil}\left(\frac{y_1 - y_{min}}{h_{hex,1}}\right)$$

$$\#hex_{ver,in,odd} = \text{ceil}\left(\frac{y_2 - y_{min}}{h_{hex,1}}\right) - \#hex_{ver,under,odd}$$

$$\#hex_{ver,under,even} = \text{trunc}\left(\frac{y_1 - y_{min}}{h_{hex,1}} + \frac{1}{2}\right)$$

$$\#hex_{ver,in,even} = \text{trunc}\left(\frac{y_2 - y_{min}}{h_{hex,1}} + \frac{1}{2}\right) - \#hex_{ver,under,even}$$

$$rec_{id,last,odd} = (2 \cdot \#hex_{hor} + 1) \cdot \#hex_{ver,under,odd}$$

$$rec_{id,last,even} = (2 \cdot \#hex_{ver,under,odd} + 1) \cdot \#hex_{hor} + \#hex_{ver,under,odd}$$

$$rec_{id,odd,i,j} = rec_{id,last,odd} + j + 2 \cdot (i - 1) \cdot \#hex_{hor}$$

$$rec_{id,even,k,l} = rec_{id,last,even} + l + 2 \cdot (k - 1) \cdot \#hex_{hor}$$

$$\{rec_{id,odd}\} = \begin{bmatrix} rec_{id,odd,1,1} & \dots & rec_{id,odd,1,\#hex_{hor,in,odd}} \\ \vdots & rec_{id,odd,i,j} & \vdots \\ rec_{id,odd,\#hex_{ver,in,odd},1} & \dots & rec_{id,odd,\#hex_{ver,in,odd},\#hex_{hor,in,odd}} \end{bmatrix}$$

$$\{rec_{id,even}\} = \begin{bmatrix} rec_{id,even,1,1} & \dots & rec_{id,even,1,\#hex_{hor,in,even}} \\ \vdots & rec_{id,even,k,l} & \vdots \\ rec_{id,even,\#hex_{ver,in,even},1} & \dots & rec_{id,even,\#hex_{ver,in,even},\#hex_{hor,in,even}} \end{bmatrix}$$

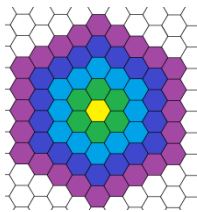
$$\{rec_{id}\} = \begin{cases} \emptyset \rightarrow x_1 > x_{max,bb} \cap x_2 < x_{min,bb} \cap y_1 > y_{max,bb} \cap y_2 < y_{min,bb} \\ \{rec_{id,odd}\} \cup \{rec_{id,even}\} \rightarrow \text{Else} \end{cases}$$

Uitvoer

$\{rec_{id}\}$

4 ae_determine_receptor_ids_from_receptor_with_radius

Functie voor het bepalen van de receptor ID's die op een gegeven afstand (in aantal hexagonen) van een andere receptor liggen. Zoals in onderstaande figuur is geschetst, ontstaat een steeds groter wordende zeshoek aan ringen om een bepaalde receptor heen. Deze functie geeft alle receptoren terug die dezelfde kleur hebben. Het algoritme geeft voor elk van de 6 zijdes eerst het meest rechte punt (vanuit het midden bekeken) terug, vervolgens één naar links etc., tot alle receptoren zijn teruggegeven. Afhankelijk of de receptor in het midden op een even of een oneven rij ligt, is de manier van bepalen waar de receptoren op de ring liggen verschillend. Hierbij wordt ook nog rekening gehouden met hexagonen die wel binnen Nederland moeten vallen.



Invoer

$rec_{id,input}, r_{input}$

Berekening

$$\#hex_{hor,for} = (rec_{id,input} - 1) \bmod (\#hex_{hor}) + 1$$

$$\#hex_{hor,after} = \#hex_{hor} - \#hex_{hor,for} + 1$$

$$rec_{id,odd,i,1} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{floor}\left(\frac{i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} + (2r_{input} - i) \cdot \#hex_{hor} - \text{floor}\left(\frac{i}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,odd,i,2} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{floor}\left(\frac{r_{input}}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} + (r_{input} - 2i) \cdot \#hex_{hor} - \text{floor}\left(\frac{r_{input}}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,odd,i,3} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{floor}\left(\frac{r_{input}-i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} - (r_{input} + i) \cdot \#hex_{hor} - \text{floor}\left(\frac{r_{input}-i}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,odd,i,4} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{ceil}\left(\frac{i}{2}\right) < \#hex_{hor,after} \\ rec_{id,input} - (2r_{input} - i) \cdot \#hex_{hor} + \text{ceil}\left(\frac{i}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,odd,i,5} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{ceil}\left(\frac{r_{input}}{2}\right) < \#hex_{hor,after} \\ rec_{id,input} - (r_{input} - 2i) \cdot \#hex_{hor} + \text{ceil}\left(\frac{r_{input}}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,odd,i,6} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{ceil}\left(\frac{r_{input}-i}{2}\right) < \#hex_{hor,after} \\ rec_{id,input} + (r_{input} + i) \cdot \#hex_{hor} + \text{ceil}\left(\frac{r_{input}-i}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,even,i,1} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{ceil}\left(\frac{i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} + (2r_{input} - i) \cdot \#hex_{hor} - \text{ceil}\left(\frac{i}{2}\right) \rightarrow Else \end{cases}$$

$$rec_{id,even,i,2} = \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap \text{ceil}\left(\frac{r_{input}}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} + (r_{input} - 2i) \cdot \#hex_{hor} - \text{ceil}\left(\frac{r_{input}}{2}\right) \rightarrow Else \end{cases}$$

$$\begin{aligned}
 rec_{id,even,i,3} &= \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap ceil\left(\frac{r_{input} - i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} - (r_{input} + i) \cdot \#hex_{hor} - ceil\left(\frac{r_{input} - i}{2}\right) \rightarrow Else \end{cases} \\
 rec_{id,even,i,4} &= \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap floor\left(\frac{i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} - (2r_{input} - i) \cdot \#hex_{hor} + floor\left(\frac{i}{2}\right) \rightarrow Else \end{cases} \\
 rec_{id,even,i,5} &= \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap floor\left(\frac{r_{input}}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} - (r_{input} - 2i) \cdot \#hex_{hor} + floor\left(\frac{r_{input}}{2}\right) \rightarrow Else \end{cases} \\
 rec_{id,even,i,6} &= \begin{cases} \emptyset \rightarrow rec_{id,odd,i,1} < 0 \cap rec_{id,odd,i,1} > rec_{id,max} \cap floor\left(\frac{r_{input} - i}{2}\right) < \#hex_{hor,for} \\ rec_{id,input} + (r_{input} + i) \cdot \#hex_{hor} + floor\left(\frac{r_{input} - i}{2}\right) \rightarrow Else \end{cases}
 \end{aligned}$$

$$\begin{aligned}
 \{rec_{id,odd}\} &= \begin{bmatrix} rec_{id,odd,1,1} & \dots & rec_{id,odd,i,6} \\ \vdots & rec_{id,odd,i,j} & \vdots \\ rec_{id,odd,r_{input},1} & \dots & rec_{id,odd,r_{input},6} \end{bmatrix} \\
 \{rec_{id,even}\} &= \begin{bmatrix} rec_{id,even,1,1} & \dots & rec_{id,even,i,6} \\ \vdots & rec_{id,even,i,j} & \vdots \\ rec_{id,even,r_{input},1} & \dots & rec_{id,even,r_{input},6} \end{bmatrix} \\
 \{rec_{id}\} &= \begin{cases} rec_{id,input} \rightarrow r_{input} = 0 \\ \{rec_{id,odd}\} \rightarrow \frac{rec_{id,input} - 1}{\#hex_{hor}} \bmod(2) = 1 \\ \{rec_{id,even}\} \rightarrow \frac{rec_{id,input} - 1}{\#hex_{hor}} \bmod(2) = 0 \end{cases}
 \end{aligned}$$

Uitvoer
 $\{rec_{id}\}$

5 ae_is_receptor_id_available_on_zoomlevel

Functie die bepaalt of een receptor ID aanwezig is op een bepaald zoomlevel.

Invoer

$rec_{id,input}, zl_{input}$

Berekening

$$\begin{aligned}
 zl_{factor} &= 2^{zl_{input}} \\
 zl_{factor-1} &= 2^{zl_{input}-1} \\
 row_{rec_{id}} &= \text{floor}\left(\frac{rec_{id,input} - 1}{\#hex_{hor}}\right) \\
 rec_{id,row} &= rec_{id,input} - row_{rec_{id}} \cdot \#hex_{hor}
 \end{aligned}$$

REC AVAILABLE

$$= \begin{cases} TRUE \rightarrow row_{rec_{id}} \text{mod}(zl_{factor}) = 0 \cup (rec_{id,row} - 1) \text{mod}(zl_{factor-1}) = 0 \\ TRUE \rightarrow row_{rec_{id}} \text{mod}(zl_{factor}) = zl_{factor-1} \cup \left(rec_{id,row} - 1 + \frac{zl_{factor-1}}{2} \right) \text{mod}(zl_{factor-1}) = 0 \\ FALSE \rightarrow Else \end{cases}$$

Uitvoer

REC AVAILABLE

6 ae_determine_coordinates_from_receptor_id

Functie voor het bepalen van de coördinaten bij een gegeven receptor ID. De x coördinaat wordt berekend door te bepalen welke receptor het is op een bepaalde rij. Verder is gegeven of de receptor op een even of oneven rij ligt, hoe breed de hexagonen zijn en wat de minimale x coördinaat is. De y coördinaat wordt bepaald door te kijken op welke rij de receptor ligt en hoe hoog de hexagonen zijn.

Invoer

$rec_{id,input}$

Berekening

$$x = x_{min,bb} + (rec_{id,input} - 1) \bmod (\#hex_{hor}) \cdot 3 \cdot r_{hex,1} + \left(\frac{rec_{id,input} - 1}{\#hex_{hor}} \right) \bmod (2) \cdot \frac{3r_{hex,1}}{2}$$
$$y = y_{min,bb} + \text{floor} \left(\frac{rec_{id,input} - 1}{\#hex_{hor}} \right) \cdot \left(\frac{h_{hex,1}}{2} \right)$$

Uitvoer

x, y

7 ae_determine_receptor_ids_from_receptor_between_radii

Deze functie bepaalt alle receptoren op zeshoeken die op een bepaalde afstand (in aantal hexagonalen) van een andere receptor liggen. Zie de uitleg bij de functie 'ae_determine_receptor_ids_from_receptor_with_radius' (factsheet G5). Deze functie roept de functie 'ae_determine_receptor_ids_from_receptor_with_radius' voor elke ring aan.

Invoer

$rec_{id,input}, r_{inner,input}, r_{outer,input}$

Berekening

$$\{rec_{id}\} = \begin{bmatrix} ae_determine_receptor_ids_from_receptor_with_radius(rec_{id,input}, r_{inner,input}) \\ \vdots \\ ae_determine_receptor_ids_from_receptor_with_radius(rec_{id,input}, r_{outer,input}) \end{bmatrix}$$

Uitvoer

$\{rec_{id}\}$

8 **ae_determine_receptor_ids_from_receptor_between_radii** (only within n2000 areas or not)

Deze functie is een meer uitgebreide vorm van de functie 'ae_determine_receptor_ids_from_receptor_between_radii' (zie hoofdstuk 7), waarbij de gebruiker aangeeft of de functie alle receptoren moet verkrijgen of alleen de receptoren binnen een Natura 2000-gebied.

Invoer

$bool_{inn2k,input}, rec_{id,input}, r_{inner,input}, r_{outer,input}$

Berekening

$$\{rec_{id}\} = \begin{cases} ae_determine_receptor_ids_from_receptor_between_radii(rec_{id,input}, r_{inner,input}, r_{outer,input}) \cap rec_{db} \\ \quad \rightarrow bool_{inn2k,input} = TRUE \\ ae_determine_receptor_ids_from_receptor_between_radii(rec_{id,input}, r_{inner,input}, r_{outer,input}) \rightarrow Else \end{cases}$$

Uitvoer

$\{rec_{id}\}$

9 ae_determine_radius_and_offset_of_outer_receptor_from_midpoint_receptor

Functie voor het bepalen van de radius en offset op deze radius van een receptor ten opzichte van een andere receptor. De radius is het aantal hexagonen dat de receptoren uit elkaar liggen. De offset is de hoeveelste receptor het is in de lijst van receptoren die teruggegeven wordt wanneer de functie 'ae_determine_receptor_ids_from_receptor_with_radius' (zie hoofdstuk 4) aangeroepen zou worden met als input de midden receptor en de straal. Hierbij wordt dus de volgorde van retourneren gebruikt uit deze functie. Deze volgorde staat beschreven bij de beschrijving van deze functie.

Invoer

$rec_{id,middle,input}, rec_{id,outer,input}$

Berekening

$$row_{rec_{id,middle}} = \text{floor}\left(\frac{rec_{id,middle,input} - 1}{\#hex_{hor}}\right) + 1$$

$$row_{rec_{id,outer}} = \text{floor}\left(\frac{rec_{id,outer,input} - 1}{\#hex_{hor}}\right) + 1$$

$$row_{diff} = row_{rec_{id,outer}} - row_{rec_{id,middle}}$$

$$rec_{id,middle,row} = (rec_{id,middle,input} - 1) \bmod (\#hex_{hor}) + 1$$

$$rec_{id,outer,row} = (rec_{id,outer,input} - 1) \bmod (\#hex_{hor}) + 1$$

$$col_{diff} = 2(rec_{id,outer,row} - rec_{id,middle,row}) + (-1)^{(row_{rec_{id,middle}} \bmod 2)} \cdot |row_{diff}| \bmod 2$$

$$r_{rec} = \begin{cases} |col_{diff}| \rightarrow |col_{diff}| > |row_{diff}| \\ \frac{|col_{diff}| + |row_{diff}|}{2} \rightarrow \text{Else} \end{cases}$$

$$part = \begin{cases} 1 \rightarrow (row_{diff} > 0 \cup col_{diff} \leq 0 \cup |col_{diff}| < |row_{diff}|) \\ 2 \rightarrow (row_{diff} > 0 \cup col_{diff} \leq 0 \cup |col_{diff}| \leq |row_{diff}|) \\ 2 \rightarrow (row_{diff} \leq 0 \cup col_{diff} < 0 \cup |col_{diff}| > |row_{diff}|) \\ 3 \rightarrow (row_{diff} \leq 0 \cup col_{diff} < 0 \cup |col_{diff}| \leq |row_{diff}|) \\ 4 \rightarrow (row_{diff} \leq 0 \cup col_{diff} \geq 0 \cup |col_{diff}| < |row_{diff}|) \\ 5 \rightarrow (row_{diff} \leq 0 \cup col_{diff} \geq 0 \cup |col_{diff}| \geq |row_{diff}|) \\ 5 \rightarrow (row_{diff} > 0 \cup col_{diff} > 0 \cup |col_{diff}| > |row_{diff}|) \\ 6 \rightarrow (row_{diff} > 0 \cup col_{diff} > 0 \cup |col_{diff}| \leq |row_{diff}|) \end{cases}$$

$$\#rounds = \begin{cases} 2r_{rec} - |row_{diff}| \rightarrow (row_{diff} > 0 \cup col_{diff} \leq 0 \cup |col_{diff}| < |row_{diff}|) \\ \frac{r_{rec} - row_{diff}}{2} \rightarrow (row_{diff} > 0 \cup col_{diff} \leq 0 \cup |col_{diff}| \leq |row_{diff}|) \\ \frac{r_{rec} - row_{diff}}{2} \rightarrow (row_{diff} \leq 0 \cup col_{diff} < 0 \cup |col_{diff}| > |row_{diff}|) \\ |row_{diff}| - r_{rec} \rightarrow (row_{diff} \leq 0 \cup col_{diff} < 0 \cup |col_{diff}| \leq |row_{diff}|) \\ 2r_{rec} - |row_{diff}| \rightarrow (row_{diff} \leq 0 \cup col_{diff} \geq 0 \cup |col_{diff}| < |row_{diff}|) \\ \frac{r_{rec} + row_{diff}}{2} \rightarrow (row_{diff} \leq 0 \cup col_{diff} \geq 0 \cup |col_{diff}| \geq |row_{diff}|) \\ \frac{r_{rec} + row_{diff}}{2} \rightarrow (row_{diff} > 0 \cup col_{diff} > 0 \cup |col_{diff}| > |row_{diff}|) \\ |row_{diff}| - r_{rec} \rightarrow (row_{diff} > 0 \cup col_{diff} > 0 \cup |col_{diff}| \leq |row_{diff}|) \end{cases}$$

$$offset_{rec} = 6 \cdot \#rounds + part$$

$$[r_{rec_{outer}}, offset_{rec_{outer}}] = \begin{cases} [r_{rec} + 1, 0] \rightarrow offset_{rec} = 6r_{rec} \cup r_{rec} > 0 \\ [r_{rec}, offset_{rec}] \rightarrow \text{Else} \end{cases}$$

Uitvoer

$r_{rec_{outer}}, offset_{rec_{outer}}$

10 ae_determine_receptor_ids_from_receptor_between_radius_and_offset

Functie voor het bepalen van een aantal receptoren vanaf een bepaalde receptor op ringen rondom een andere receptor waarbij de gebruiker aangeeft dat hij alle receptoren wil krijgen of alleen de receptoren binnen een Natura 2000-gebied. Het algoritme begint met het aanroepen van de functie 'ae_determine_radius_and_offset_of_outer_receptor_from_midpoint_receptor' (zie hoofdstuk 9) om te bepalen wat de binnenste zeshoek is waarvan de receptoren teruggegeven moeten worden en vanaf welke receptor in de lijst dit is. Vervolgens worden net zo lang receptoren uit ringen toegevoegd tot het aantal limit_input bereikt is. Wanneer de gebruiker heeft aangegeven dat het gaat om alleen de receptoren binnen een Natura 2000-gebied, dan worden een limit_input aantal receptoren teruggegeven.

Invoer

$rec_{id,inner,input}$, $rec_{id,outer,input}$, $limit_{input}$, $bool_{inn2k,input}$

Berekening

$$\begin{aligned}
 [r_{rec}, offset_{rec}] &= ae_determine_radius_and_o \dots dpoint_receptor(rec_{id,inner,input}, rec_{id,outer,input}) \\
 \{rec_{id,rad,all}\} &= ae_determine_receptor_ids_from_receptor_with_radius(rec_{id,inner,input}, r_{rec}) \\
 \{rec_{id,rad,all,1}\} &= \begin{bmatrix} rec_{id,rad,all,offset_{rec}} \\ \vdots \\ rec_{id,rad,all,last} \end{bmatrix} \\
 \{rec_{id,rad,all,i}\} &= \{rec_{id,rad+i-1,all}\} \\
 \{rec_{id,output,all}\} &= \left. \begin{bmatrix} \{rec_{id,rad,all,1}\} \\ \vdots \\ \{rec_{id,rad,all,i}\} \\ \vdots \\ \{rec_{id,rad+k,all,j}\} \end{bmatrix} \right\} limit_{input} \\
 \{rec_{id,rad,n2k}\} &= ae_determine_receptor_ids_from_receptor_with_radius(rec_{id,inner,input}, r_{rec}) \cap rec_{db} \\
 \{rec_{id,rad,n2k,1}\} &= \begin{bmatrix} rec_{id,rad,n2k,offset_{rec}} \\ \vdots \\ rec_{id,rad,n2k,last} \end{bmatrix} \\
 \{rec_{id,rad,n2k,i}\} &= \{rec_{id,rad+i-1,n2k}\} \\
 \{rec_{id,output,n2k}\} &= \left. \begin{bmatrix} \{rec_{id,rad,n2k,1}\} \\ \vdots \\ \{rec_{id,rad,n2k,i}\} \\ \vdots \\ \{rec_{id,rad+k,n2k,j}\} \end{bmatrix} \right\} limit_{input} \\
 \{rec_{id,output}\} &= \begin{cases} \{rec_{id,output,all}\} \rightarrow bool_{inn2k,input} = FALSE \\ \{rec_{id,output,n2k}\} \rightarrow bool_{inn2k,input} = TRUE \end{cases}
 \end{aligned}$$

Uitvoer

$\{rec_{id,output}\}$