

```

> restart,
>
>
> # Set parameters which are inferred directly from literature
> delta1 := 0.25 :
> delta2 := 0.25 :
> k1 := 65 :
> k2 := k1 :
> rs := 0.04 :
> rg := 0.04 :
> ag := 0.14 :
>
>
> # Define quasi-2d potentials
> Qs := unapply( Rs·evalf( Int( exp( - sqrt(x^2 + y^2) / rs ), y=-infinity..infinity ) ), x ) :
> Qg := unapply( evalf( Rg·evalf( Int( exp( - sqrt(x^2 + y^2) / rg ), y=-infinity..infinity ) ) - Ag
    ·evalf( Int( exp( - sqrt(x^2 + y^2) / ag ), y=-infinity..infinity ) ) ), x ) :
>
>
> # Construct imaginary density field
> rho0 := k1 :
> rho := x→rho0·Heaviside(x) :
>
>
> # Compute Rs from solitary locust at edge of swarm
> vsmax := -216 :
> eq1 := rho0·(-Qs(0)) = vsmax;
    eq1 := -65 Rs ( ∫_{Float(-∞)}^{Float(∞)} e^{-25.00000000 √y^2} dy ) = -216 (1)
> Rs := solve(eq1, Rs);
    Rs := 41.53846154 (2)
>
>
> # Compute Rg,Ag from gregarious locust at edge of swarm and then at short distance away
> vgmax := 216 :
> vgmin := 144 :
> eq2 := evalf(rho0·(-Qg(0)) = vgmax);
    eq2 := -5.200000000 Rg + 18.20000000 Ag = 216. (3)

```

```
> eq3 := evalf(rho0*(-Qg(-0.14)) = vgmin);  
eq3 := -0.4047569512 Rg + 10.95471159 Ag = 144. (4)
```

```
> sol := solve({eq2, eq3}, {Rg, Ag});  
sol := {Ag = 13.33468114, Rg = 5.132922455} (5)
```

```
> assign(sol) :
```

```
>  
>
```

```
> # Check for correct limiting behavior of Qg
```

```
> # For repulsion at small length scales, this quantity should be positive
```

```
>  $\frac{Rg}{rg} - \frac{Ag}{ag}$ ;  
33.07533897 (6)
```

```
> # For attraction at long length scales, this quantity should be negative
```

```
>  $Rg \cdot rg^2 - Ag \cdot ag^2$ ;  
-0.2531470744 (7)
```

```
>  
>
```