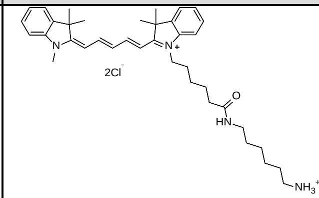
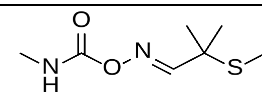
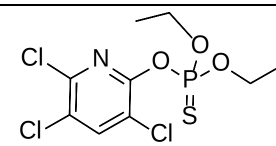
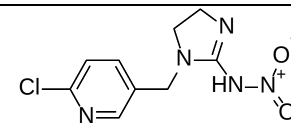
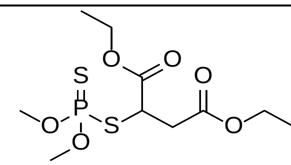
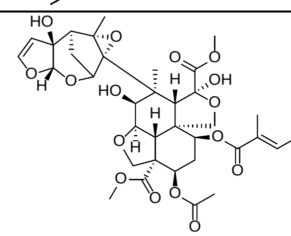
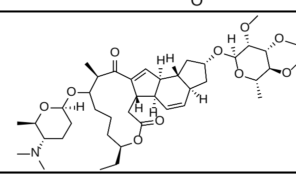
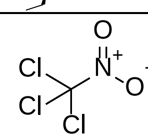
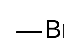
nature nanotechnology
SUPPLEMENTARY INFORMATION
https://doi.org/10.1038/s41565-019-0453-7
In the format provided by the authors and unedited.

Soil mobility of synthetic and virus-based model nanopesticides

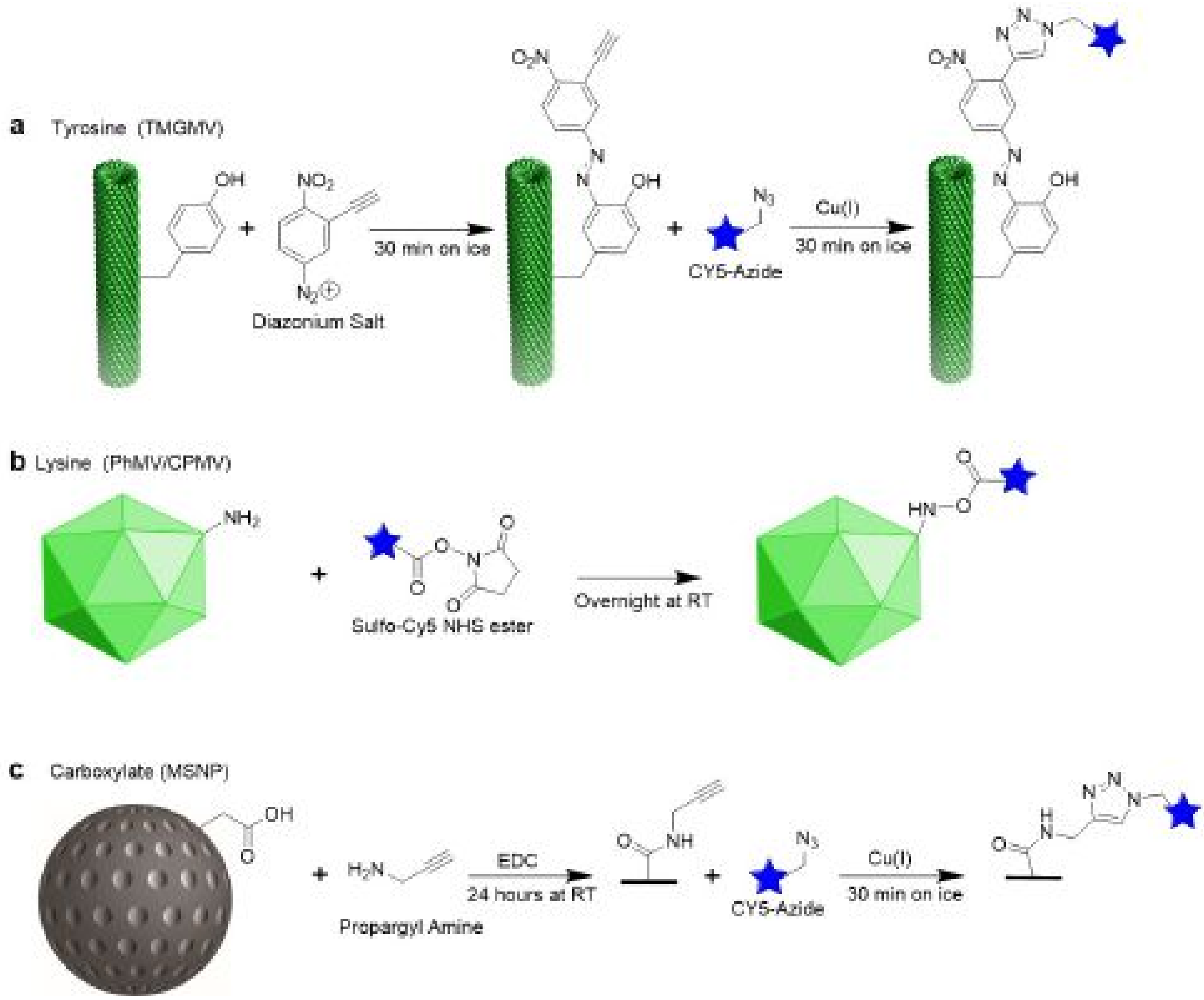
Paul L. Chariou [1,2], Alan B. Dogan [2], Alexandra G. Welsh [3], Gerald M. Saidel [2], Harihara Baskaran [3] and Nicole F. Steinmetz [1,2,4,5,6]*

[1]Department of Bioengineering, University of California-San Diego, La Jolla, CA, USA. [2]Department of Biomedical Engineering, Case Western Reserve University, Cleveland, OH, USA. [3]Department of Chemical and Biomolecular Engineering, Case Western Reserve University, Cleveland, OH, USA. [4]Department of NanoEngineering, University of California-San Diego, La Jolla, CA, USA. [5]Moores Cancer Center, University of California-San Diego, La Jolla, CA, USA. [6]Department of Radiology, University of California-San Diego, La Jolla, CA, USA. *e-mail: nsteinmetz@ucsd.edu

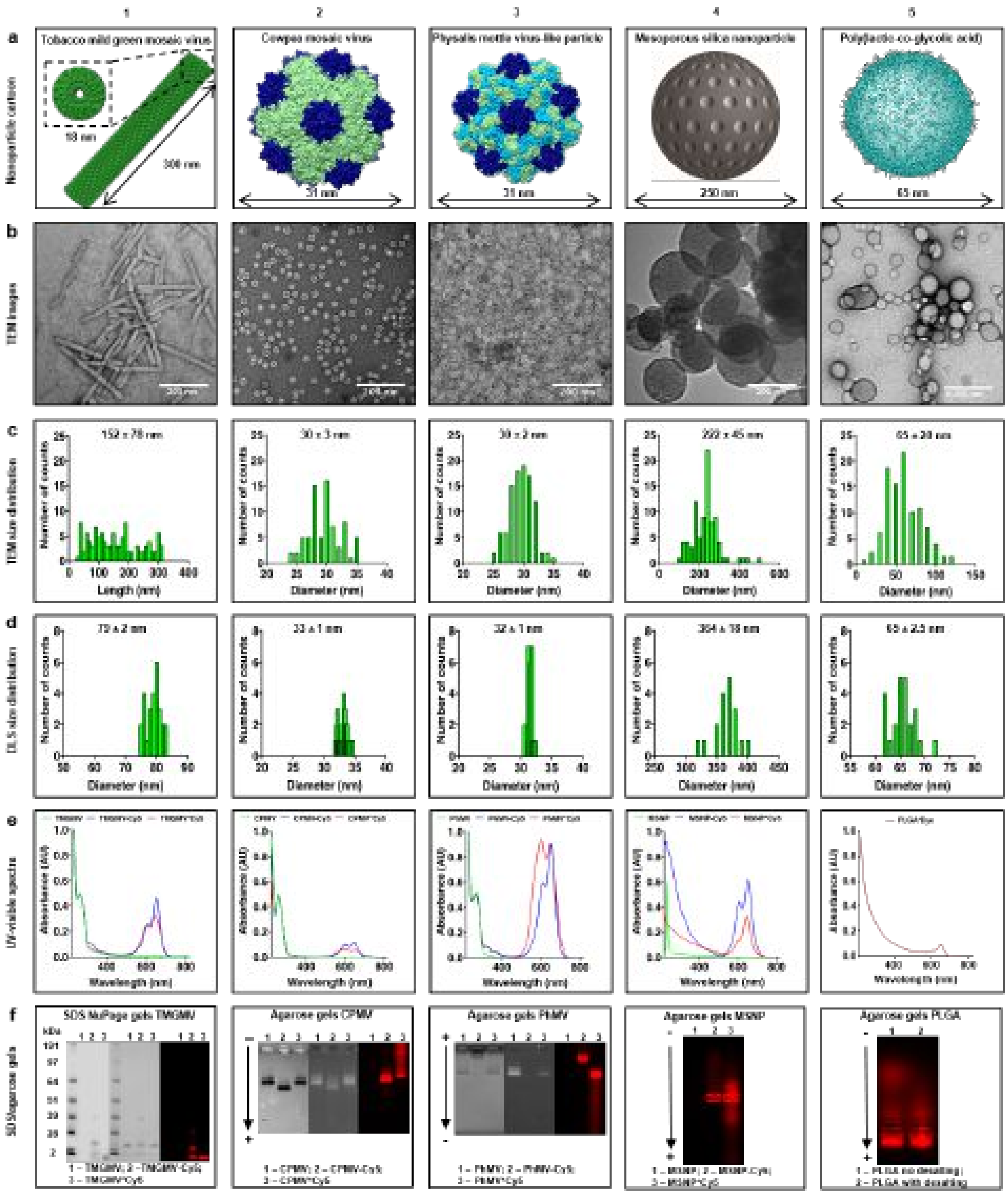
Supplementary Table 1 | Comparison of conventional pesticides to the model drug Cy5

Root pests	Pesticide	**Log P (oct/aqu)	Molecular weight (g/mol)	Chemical structure
Model pesticide	Cyanine 5	3.2	747	
Nematode	Aldicarb	1.13	190.3	
	Chlorpyrifos	4.96	350	
Beetle grubs	Imidacloprid	0.57	255.6	
	Melathion	2.36	330.4	
Moth caterpillars and root aphids	Azadirachtin	1.09	720.7	
	Spinosad	2.8	731.9	
Root fungus	Chloropicrin	2.09	164.4	
	Methyl bromide	3.42	94.9	

<u> Log P</u>: Thepartition-coefficient** (***P***) is the concentration ratio of a molecule in a mixture of two immiscible phases: water and octanol. It measures the degree of hydrophobicity or hydrophilicity.



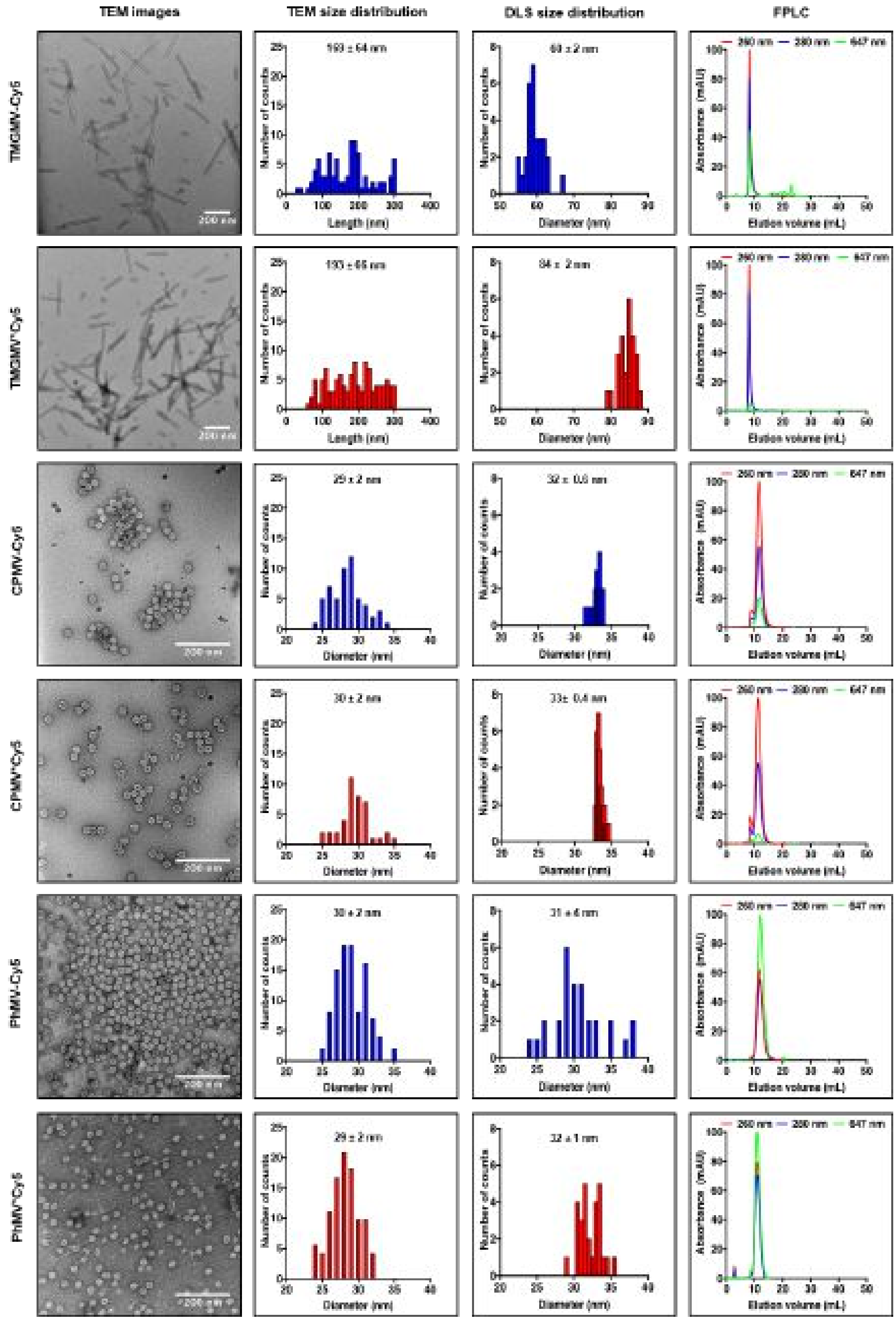
Supplementary Fig. 1 | Cy5 conjugation to TMGMV, PhMV, CPMV and MSNP. The schematics show chemical conjugation of Cy5 to **a,** the surface-exposed tyrosine residues of TMGMV using diazonium chemistry followed by click chemistry, **b,** the surface exposed lysine residues of PhMV and CPMV using NHS chemistry, and **c,** the carboxylate groups of MSNP via EDC and click chemistry.



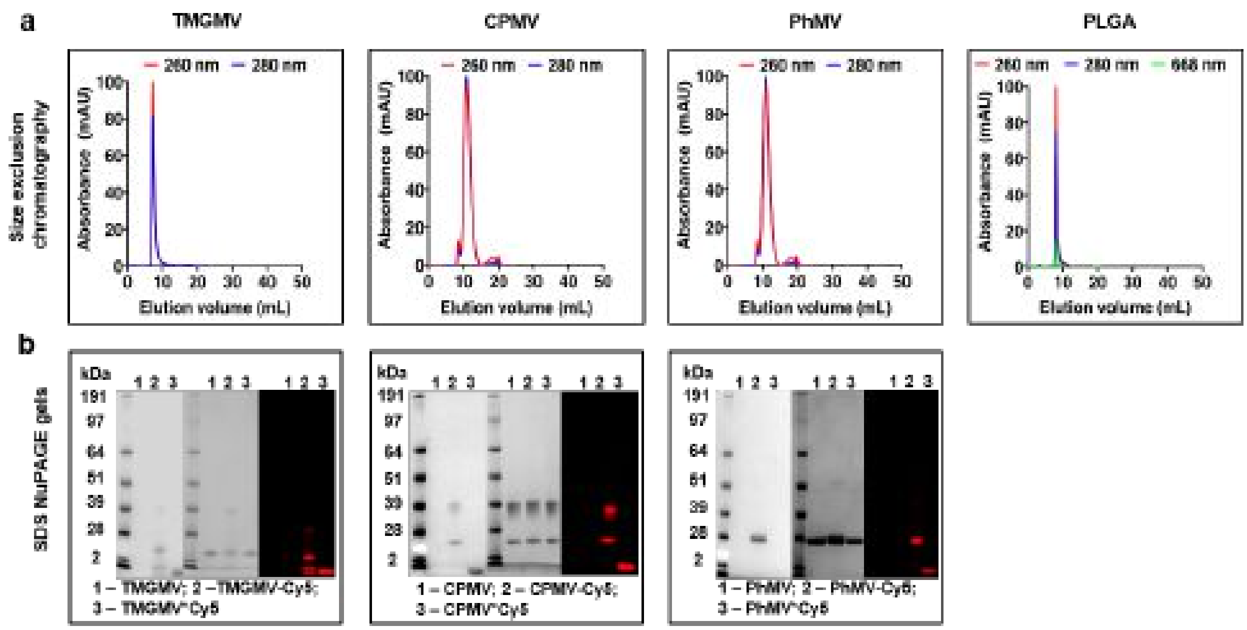
Supplementary Fig. 2 | Nanoparticle characterization. a, Simplified representation of (1) TMGMV (PDB ID: 1VTM), (2) CPMV (PDB ID: 1NY7), (3) PhMV (PDB ID: 1QJZ), (4) MSNP, and (5) PLGA. **b,** Corresponding TEM images obtained using a Tecnai F-30 transmission electron microscope. Scale bar = 200 nm. **c,** Size distribution analysis of the TEM images. **d,** Particle size distribution obtained by dynamic light scattering (DLS). **e,** UV/Vis spectra of bare nanoparticles (green), Cy5-conjugated nanoparticles (blue), and Cy5-infused nanoparticles (red). Spectra were normalized to the 260 nm wavelength peak. **f,** SDS-PAGE analysis of the different TMGMV formulations. Left to right gels were imaged under white light before staining, white light after staining, and under red fluorescence, respectively. In addition, agarose gels of the CPMV, PhMV, MSNP and PLGA formulations are shown. Left to right agarose gels were imaged under white light after staining, UV light, and under red fluorescence.

Characterization results:

TEM imaging revealed monodisperse CPMV (30 ± 3 nm) and PhMV (30 ± 2 nm) samples, but polydisperse TMGMV (152 ± 78 nm), MSNP (222 ± 45 nm), and PLGA (65 ± 20 nm) samples (Fig. S2B-C). Plant viruses are naturally produced as identical copies and therefore are highly monodisperse. This level of quality control has yet to be achieved using synthetic nanoparticles such as MSNP and PLGA. The polydispersity of TMGMV has been reported[1] and is attributed to greater particle instability during virus production and purification and due to the TEM grid preparation, when the particles may break. DLS was also used to measure the hydrodynamic radii of TMGMV (79 ± 2 nm), CPMV (33 ± 1 nm), PhMV (32 ± 2 nm), MSNP (364 ± 18 nm), and PLGA (65 ± 2.5 nm), in agreement with the TEM data (Fig. S2d). The hydrodynamic radius of TMGMV is inaccurate because of the rod-like shape of the particle, but the data can still be used to semi-quantify the degree of polydispersity of the TMGMV sample. We attribute the increased hydrodynamic radius of MSNP to its physicochemical properties (many carboxylate residues on the surface, which may promote the formation of a water shell around the nanoparticle). Particles were further characterized post-conjugation or after the encapsulation of Cy5. TEM and DLS results revealed no apparent differences in size and polydispersity between the nanoparticles with and without Cy5 (Fig. S3). The dye loading efficiency was quantified by UV/Vis spectroscopy using the Beer–Lambert law and specific extinction coefficients (Fig. S2e). Overall, the dye loading efficiency (nmol of dye per mg of nanoparticle) was higher when the fluorophore was conjugated to the nanoparticle (denoted as nanoparticle-Cy5) rather than passively encapsulated (denoted as nanoparticle*Cy5). Specifically, the loading efficiencies of TMGMV-Cy5 (9.9 nmol mg^{-1} or 390 dyes per TMGMV), CPMV-Cy5 (6.2 nmol mg^{-1} or 35 dyes per CPMV), PhMV-Cy5 (12.7 nmol mg^{-1} or 60 dyes per PhMV), and MSNP-Cy5 (6.4 nmol mg^{-1}) were higher than corresponding encapsulated formulations TMGMV*Cy5 (5.3 nmol mg^{-1} or 210 dyes per TMGMV), CPMV*Cy5 (2.3 nmol mg^{-1} or 15 dyes per CPMV), PhMV*Cy5 (11.7 nmol mg^{-1} or 55 dyes per PhMV), MSNP*Cy5 (4.3 nmol mg^{-1}), and PLGA*Dye (1.2 nmol mg^{-1}). The loading efficiency can be ranked from highest to lowest as follows: PhMV > TMGMV > CPMV > MSNP > PLGA. Agarose gel electrophoresis, SDS-PAGE and SEC (Fig. S4) confirmed the covalent attachment or encapsulation of Cy5, as indicated by the presence of fluorescence under red light (Figs S2f and S4). SDS-PAGE only works with protein samples, so PLGA and MNSP were characterized in agarose gels only. Due to its shape and size, TMGMV has no mobility in agarose gels and therefore was only characterized by SDS-PAGE. On the agarose gels, free Cy5 moves toward the anode due to its overall positive charge. Because of their net electronegativity, CPMV, PLGA and MSNP move toward the cathode, whereas the positively charged PhMV particles move toward the anode. The Cy5-conjugated nanoparticle samples showed no apparent sign of free fluorophores. On the other hand, the formulation containing Cy5 passively encapsulated in the nanoparticles revealed that a portion of the Cy5 was indeed released during gel electrophoresis. In addition to the red light, agarose gels loaded with viruses were imaged under white light after staining with Commassie Brilliant Blue, and under UV light to check for the presence of RNA. Both PhMV and CPMV samples revealed the presence of encapsulated RNA. The presence of RNA in PhMV is the result of “junk” RNA encapsulation during the capsid self-assembly in the bacterial culture. Denaturing SDS-PAGE gels further confirmed the covalent attachment or encapsulation of Cy5 (Figs S2f and S4). The TMGMV and PhMV coat proteins are ~17 kDa and ~26 kDa, respectively. CPMV is composed of 60 coat proteins, each comprising a small (24 kDa) and a large (42kD) subunit. Cy5 has a molecular weight of ~650 Da, which is too small to cause a visible shift of the coat protein electrophoretic mobility. Nonetheless, the covalently bound Cy5 co-localize with the coat proteins of TMGMV, PhMV and CPMV, whereas the encapsulated dye travels freely to the anode. In the Cy5 conjugated virus samples, a portion of the Cy5 travelled freely to the anode. Because the samples were meticulously desalted prior to the experiment to remove any remaining free Cy5, this result indicates that the dye molecules detached from the coat proteins during sample preparation.



Supplementary Fig. 3 | Characterization of Cy5-labelled viruses.



Supplementary Fig. 4 | Size exclusion chromatography and SDS-PAGE analysis of Cy5-labelled viruses. a, Size exclusion chromatography of TMGMV, CPMV, PhMV and PLGA. **b,** SDS-PAGE analysis of viruses. From left to right, gels were imaged under while light before staining, after staining with Coomassie Brilliant Blue, and under red light.

Supplementary Table 2 | Soil composition.

Analysis was contracted to Western Laboratories, Parma, ID.

ELEMENT	ANSWER	INTERP	SHOULD BE
pH-Soil	6.7	Neutral Soil	
pH-SMP			
Soluble Salts	0.30	Optimum	< 1.5
% Lime	0	No lime	
% Organic Matter	36.55	Very High	
Nitrates-ppm	27	Optimum	10 - 35
Ammonium-ppm	6	Optimum	5 +
Phosphorus-ppm	7	Very Low	25 - 40
Phos-ppm-Bray			50 - 100
Potassium-ppm	125	Low	300 +

ELEMENT	ANSWER	INTERP	SHOULD BE
Sulfur-ppm	12	Low	20 +
Calcium-ppm	3567	High	1,800 +
Magnesium-ppm	674	Very High	250 +
Sodium-ppm	29	Optimum	< 225
Zinc-ppm	0.5	Very Low	1.0 - 3.0
Copper-ppm	0.8	Low	0.8 - 2.5
Manganese-ppm	4	Low	6 - 30
Iron-ppm	11	Medium	25 +
Boron-ppm	0.3	Very Low	0.7 - 1.5
TBS%	0		

Texture		Water Holding Capacity/foot		Bulk Density	

Cation Exchange Capacity - CEC	7	P Index	
Percent Base Saturation	98		

BASES	IDEAL	YOURS
Calcium-% of CEC	65-80	73
Magnesium-% of CEC	10-20	23
Potassium-% of CEC	2-6	1
Sodium-% of CEC (ESP)	< 5	1
Hydrogen-% of CEC	< 15	2

	NO3 ppm	NH4 ppm
1 Ft	27	6
2 Ft		
3 Ft		
Total N PPM		33
Lbs N / Acre		99

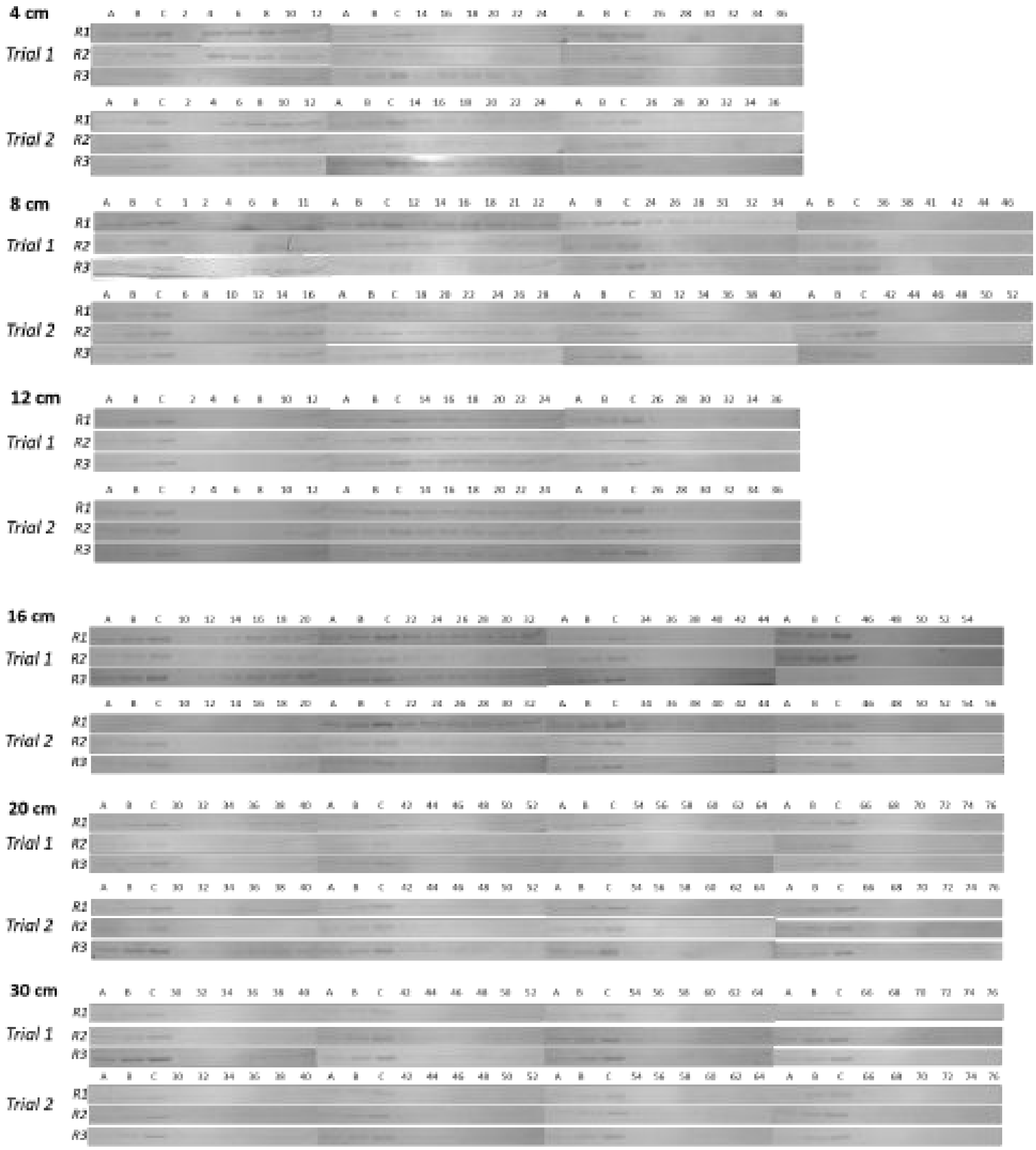
Ratio	Ideal	Yours	Evaluation	Recommendations
Ca:Mg	6-20:1	5 :1	Low	Watch Ca
Ca:K pH >7	15:1	:1		
Ca:K pH <7	10:1	29 :1	Low	
Ca:P pH >7	100:1	:1		
Ca:P pH <7	40:1	510:1	High	Watch P
P:Zn	15:1	14 :1	OK	
P:Mn	4:1	2:1	OK	
P:Cu	25:1	9 :1	OK	
Zn:Cu	3:1	1 :1	OK	
Mn:Zn	3:1	8 :1	High	Watch Zn
Mn:Cu	7:1	5:1	OK	
K:B	200:1	417 :1	High	Watch B
Mg:K	2:1	5 :1	High	Watch K

Fertilizer Suggestions in Pounds per Acre for the whole season		
Crop		
Yield Goal		
Past Crop		
Acres		
Nitrogen		
Phosphate		
Add Phos for P INDEX		
Potash		
P.F. Sulfur		
Elemental Sulfur		
Gypsum		
Lime		
Dolomite		
Magnesium		
Zinc		
Manganese		
Copper		

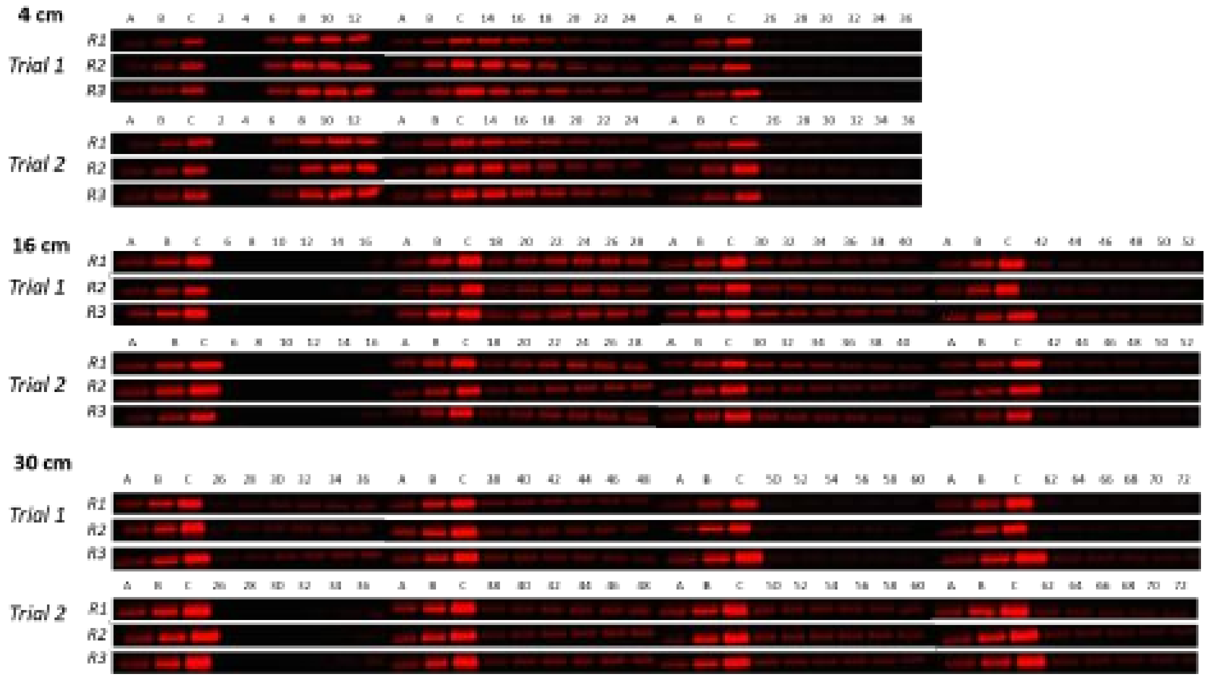
Supplementary Table 3 | Soil particle size distribution and adsorption surface as a function of particle volume.

Adsorption surface per particle volume ϕ.			Distribution				
Particle Size (µm)	Surface area (µm2)	Volume (µm3)	Read 1	Read 2	Average	Φ (µm^-1)	Weighted average Φ (µm^-1)
1000	3140000	523333333	0.72	*1.34*	1.03	0.0060	0.0000618
500	785000	65416667	8.29	*8.60*	8.45	0.0120	0.0010134
250	196250	8177083	14.31	*13.69*	14.00	0.0240	0.00336
200	125600	4186667	5.46	*5.34*	5.40	0.0300	0.00162
150	70650	1766250	7.55	*7.47*	7.51	0.0400	0.003004
125	49063	1022135	5.03	*5.00*	5.02	0.0480	0.0024072
100	31400	523333	6.28	*6.27*	6.28	0.0600	0.003765
75	17663	220781	7.92	*7.92*	7.92	0.0800	0.006336
62.5	12266	127767	4.68	*4.70*	4.69	0.0960	0.0045024
50	7850	65417	5.27	*5.29*	5.28	0.1200	0.006336
40	5024	33493	4.72	*4.74*	4.73	0.1500	0.007095
30	2826	14130	5.29	*5.29*	5.29	0.2000	0.01058
20	1256	4187	6.18	*6.14*	6.16	0.3000	0.01848
19	1134	3590	0.69	*0.68*	0.69	0.3158	0.002163158
18	1017	3052	0.69	*0.68*	0.69	0.3333	0.002283333
17	907	2571	0.71	*0.71*	0.71	0.3529	0.002505882
16	804	2144	0.73	*0.72*	0.73	0.3750	0.00271875
15	707	1766	0.75	*0.74*	0.75	0.4000	0.00298
14	615	1436	0.77	*0.76*	0.77	0.4286	0.003278571
13	531	1150	0.79	*0.79*	0.79	0.4615	0.003646154
12	452	904	0.81	*0.81*	0.81	0.5000	0.00405
11	380	697	0.85	*0.85*	0.85	0.5455	0.004636364
10	314	523	0.88	*0.88*	0.88	0.6000	0.00528
9	254	382	0.92	*0.91*	0.92	0.6667	0.0061
8	201	268	0.97	*0.96*	0.97	0.7500	0.0072375
7	154	180	1.02	*1.02*	1.02	0.8571	0.008742857
6	113	113	1.09	*1.09*	1.09	1.0000	0.0109
5	79	65	1.15	*1.14*	1.15	1.2000	0.01374
4	50	33	1.22	*1.22*	1.22	1.5000	0.0183
3	28	14	1.27	*1.26*	1.27	2.0000	0.0253
2	13	4	1.32	*1.32*	1.32	3.0000	0.0396
1.5	7	2	0.67	*0.67*	0.67	4.0000	0.0268
1	3	1	0.65	*0.66*	0.66	6.0000	0.0393
0.5	1	0	0.35	*0.35*	0.35	12.0000	0.042
0.01	0	0	0.00	*0.00*	0.00	600.0000	0
0	0	0	0.00	*0.00*	0.00		0
ΣSum			100.0	*100.0*	100.0		0.34012337
Sand			60.2	*60.3*			
Silt			38.1	*38.0*			
Clay			1.7	*1.7*			
Σ√			100.0	*100.0*			
NRCS Soil Texture			sandy loam	*sandy loam*			

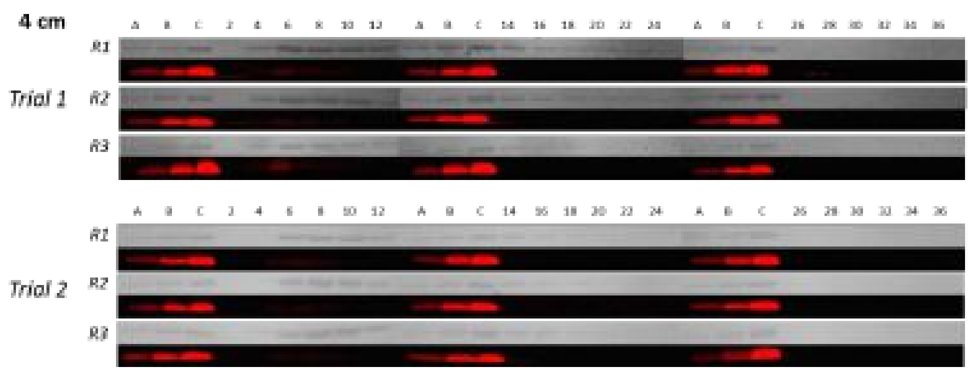
TMGMV


4 cm
Trial 1
Trial 2
8 cm
Trial 1
Trial 2
12 cm
Trial 1
Trial 2
16 cm
Trial 1
Trial 2
20 cm
Trial 1
Trial 2
30 cm
Trial 1
Trial 2

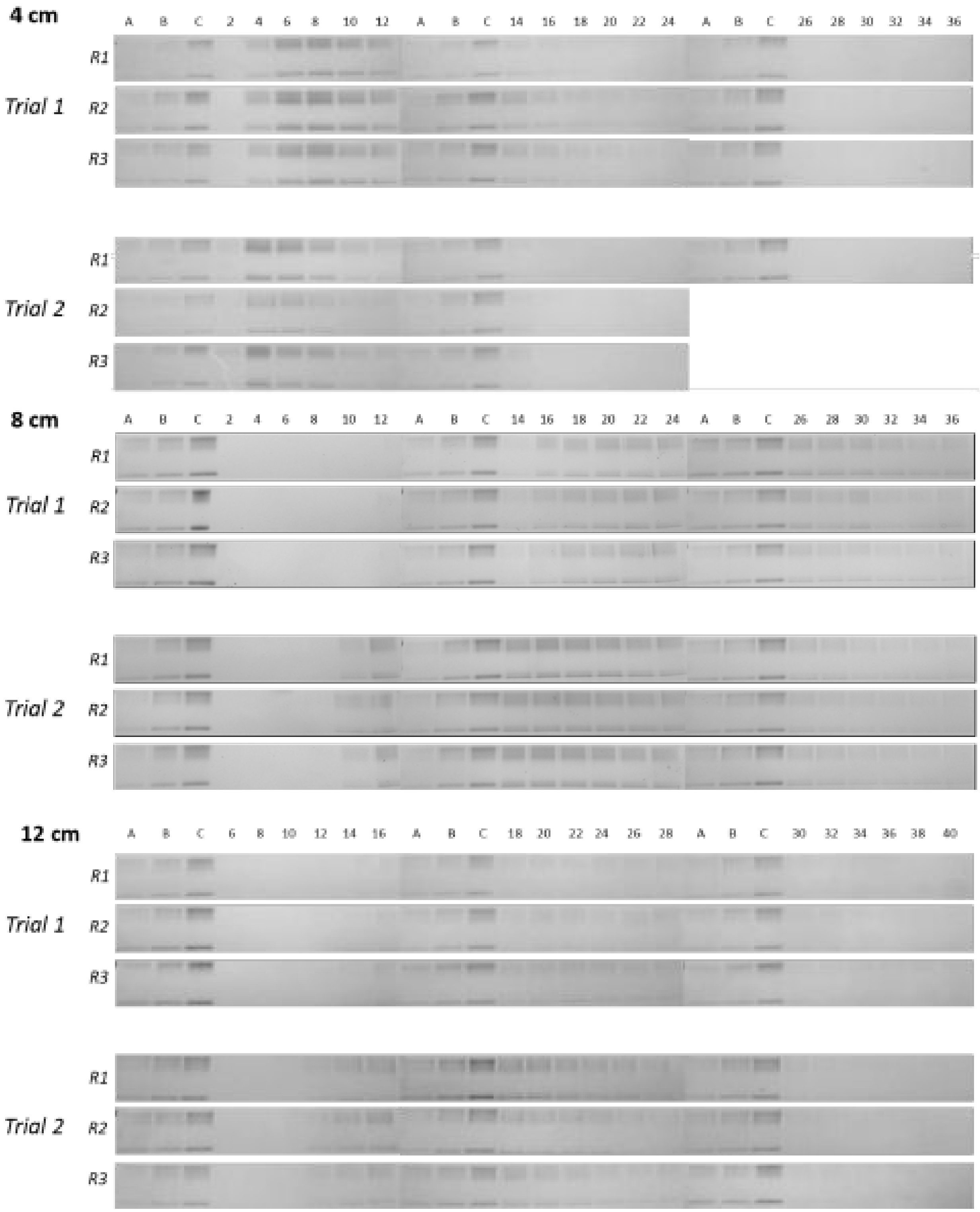
TMGMV-Cy5

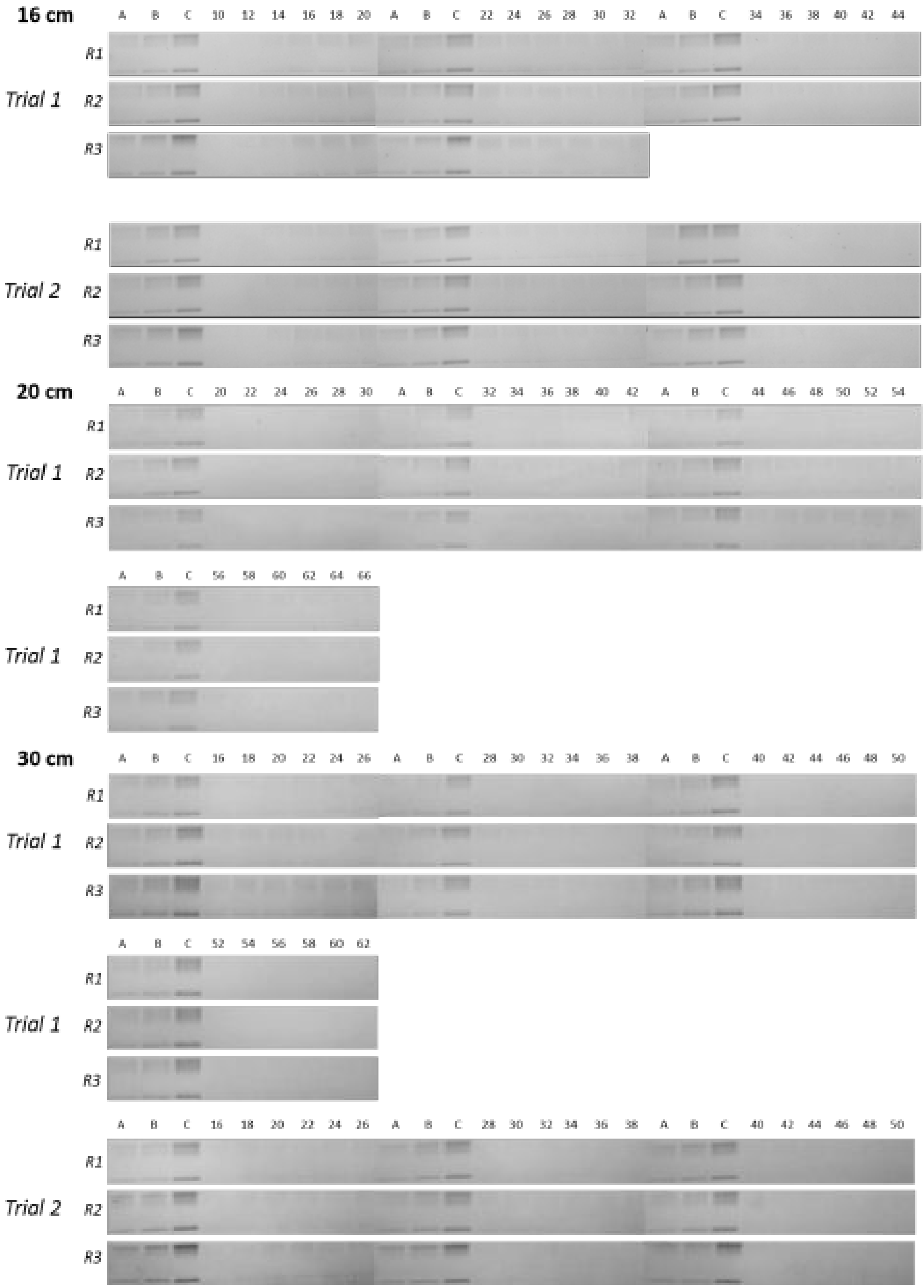


TMGMV*Cy5

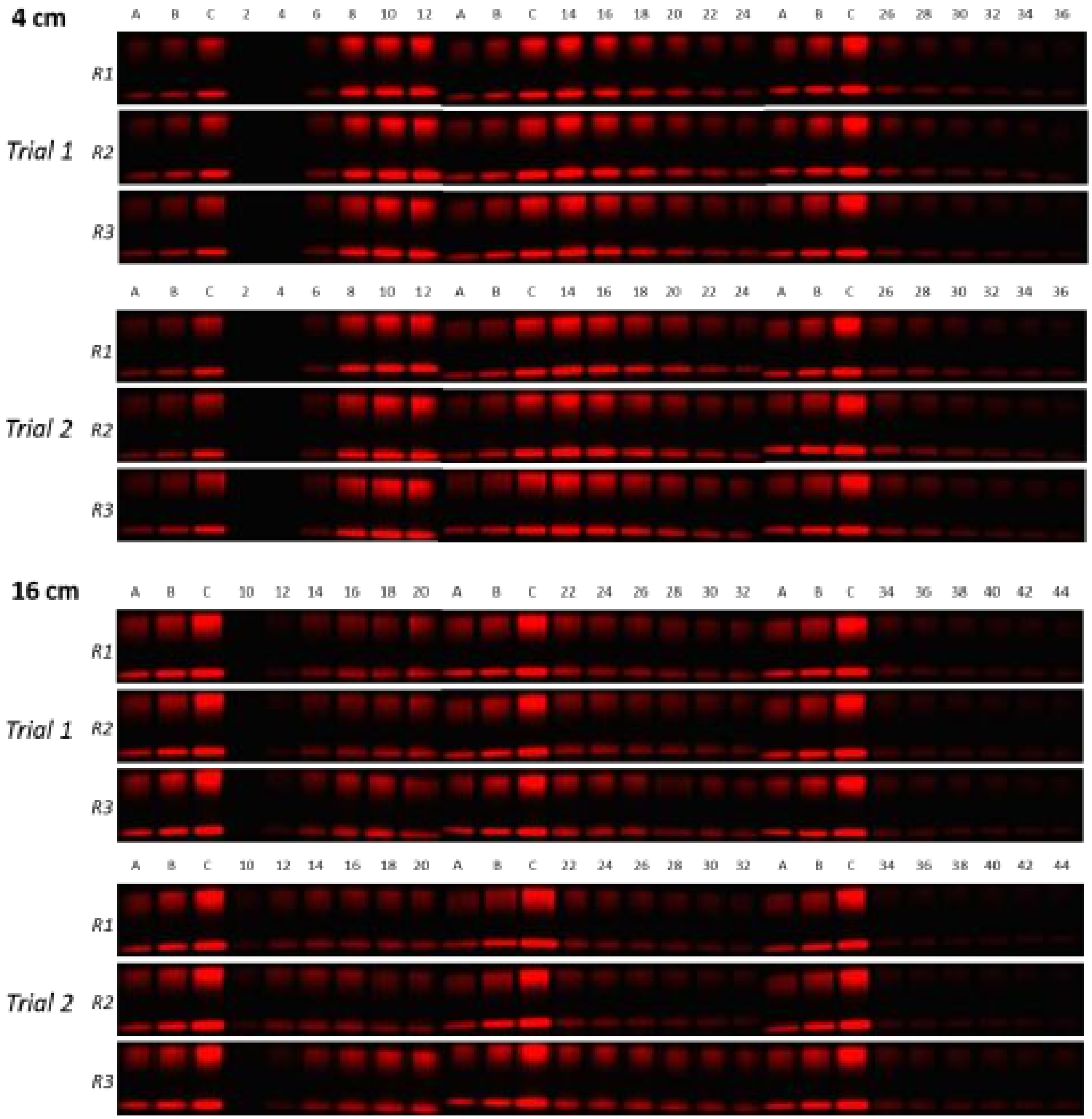


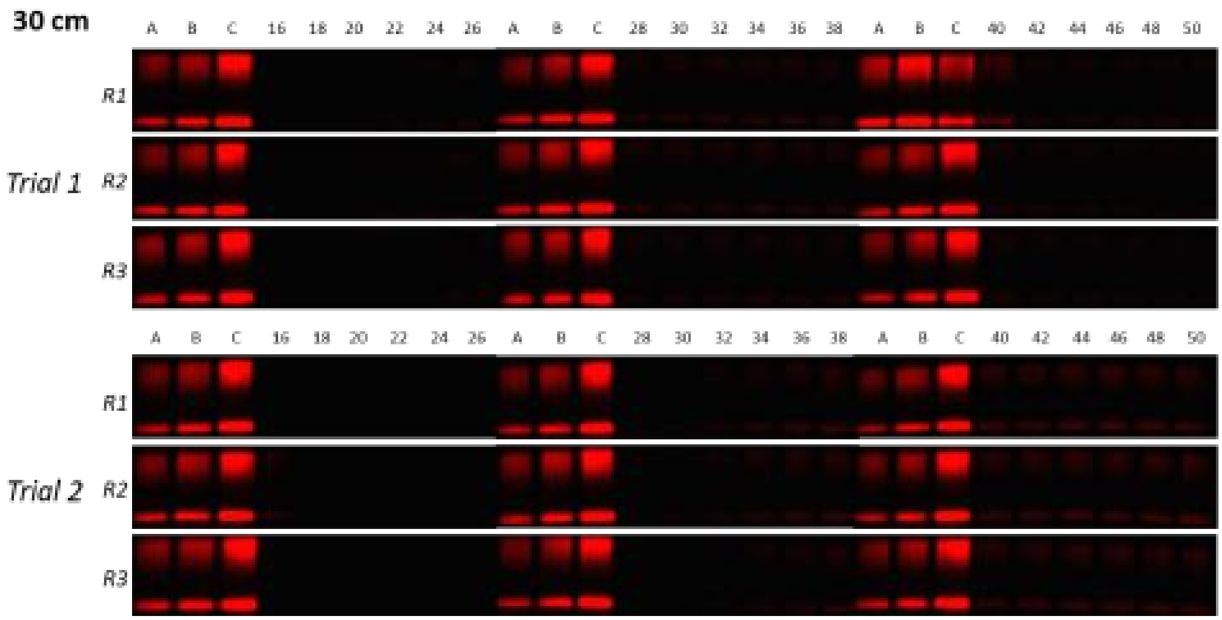
CPMV


4 cm
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24 A B C 26 28 30 32 34 36
Trial 1
R1
R2
R3
Trial 2
R1
R2
R3
8 cm
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24 A B C 26 28 30 32 34 36
Trial 1
R1
R2
R3
Trial 2
R1
R2
R3
12 cm
A B C 6 8 10 12 14 16 A B C 18 20 22 24 26 28 A B C 30 32 34 36 38 40
Trial 1
R1
R2
R3
Trial 2
R1
R2
R3

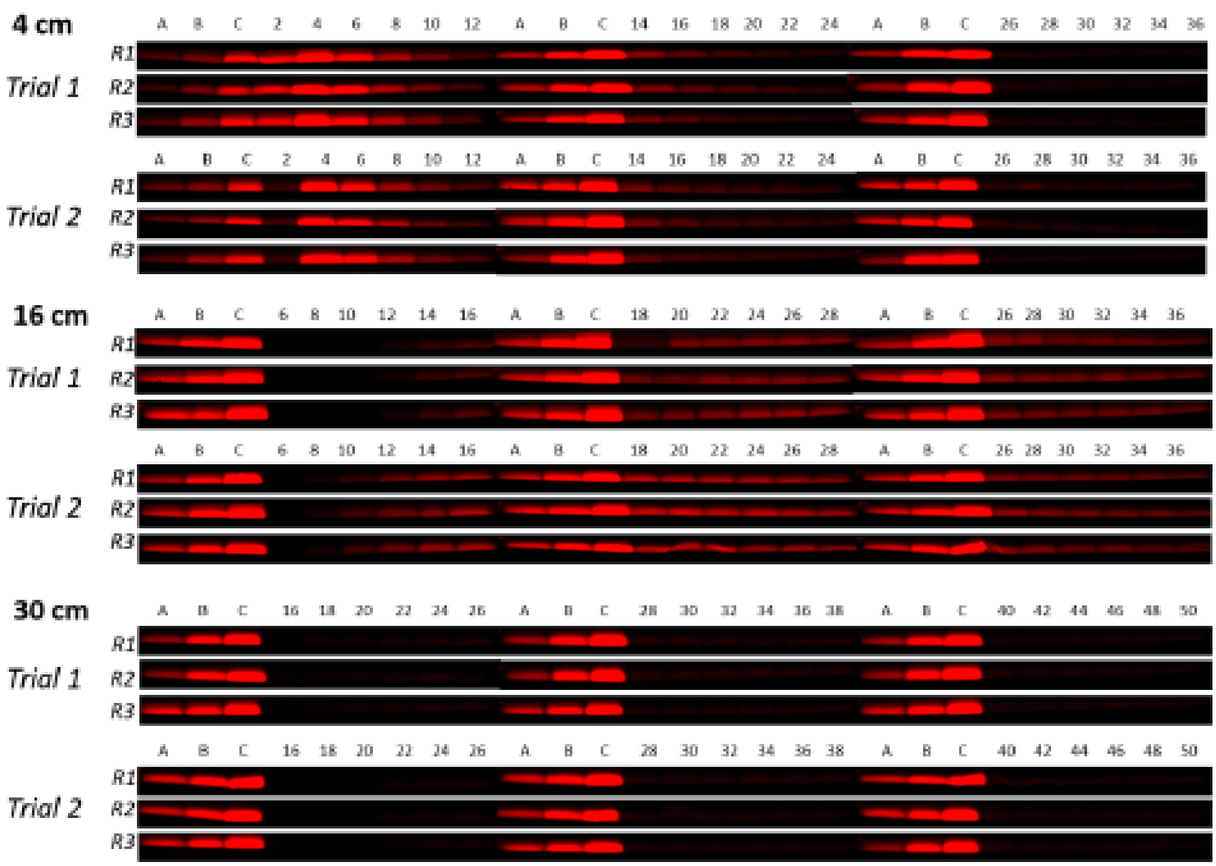

16 cm
A B C 10 12 14 16 18 20 A B C 22 24 26 28 30 32 A B C 34 36 38 40 42 44
Trial 1
R1
R2
R3
Trial 2
R1
R2
R3
20 cm
A B C 20 22 24 26 28 30 A B C 32 34 36 38 40 42 A B C 44 46 48 50 52 54
Trial 1
R1
R2
R3
A B C 56 58 60 62 64 66
Trial 1
R1
R2
R3
30 cm
A B C 16 18 20 22 24 26 A B C 28 30 32 34 36 38 A B C 40 42 44 46 48 50
Trial 1
R1
R2
R3
A B C 52 54 56 58 60 62
Trial 1
R1
R2
R3
A B C 16 18 20 22 24 26 A B C 28 30 32 34 36 38 A B C 40 42 44 46 48 50
Trial 2
R1
R2
R3

CPMV-Cy5

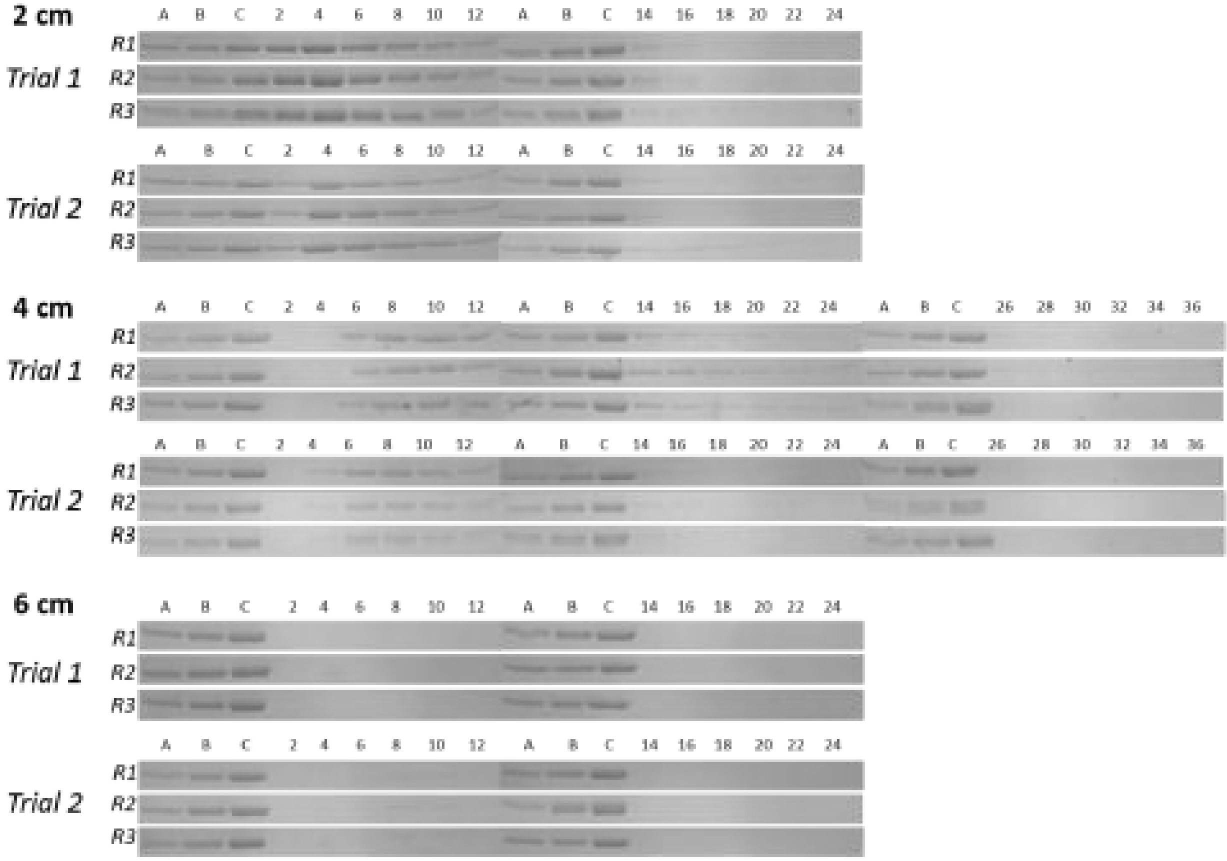

4 cm
Trial 1
Trial 2
R1
R2
R3
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24 A B C 26 28 30 32 34 36
16 cm
A B C 10 12 14 16 18 20 A B C 22 24 26 28 30 32 A B C 34 36 38 40 42 44


30 cm
A B C 16 18 20 22 24 26 A B C 28 30 32 34 36 38 A B C 40 42 44 46 48 50
Trial 1
R1
R2
R3
Trial 2
R1
R2
R3

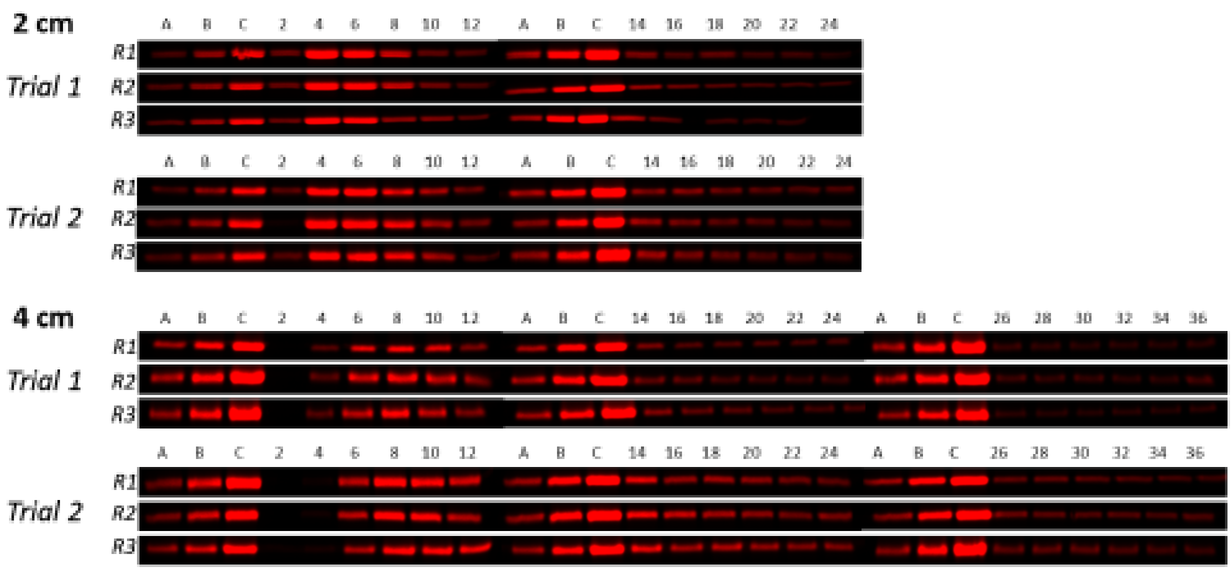
CPMV*Cy5


4 cm
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24 A B C 26 28 30 32 34 36
Trial 1
Trial 2
R1
R2
R3
16 cm
A B C 6 8 10 12 14 16 A B C 18 20 22 24 26 28 A B C 26 28 30 32 34 36
30 cm
A B C 16 18 20 22 24 26 A B C 28 30 32 34 36 38 A B C 40 42 44 46 48 50

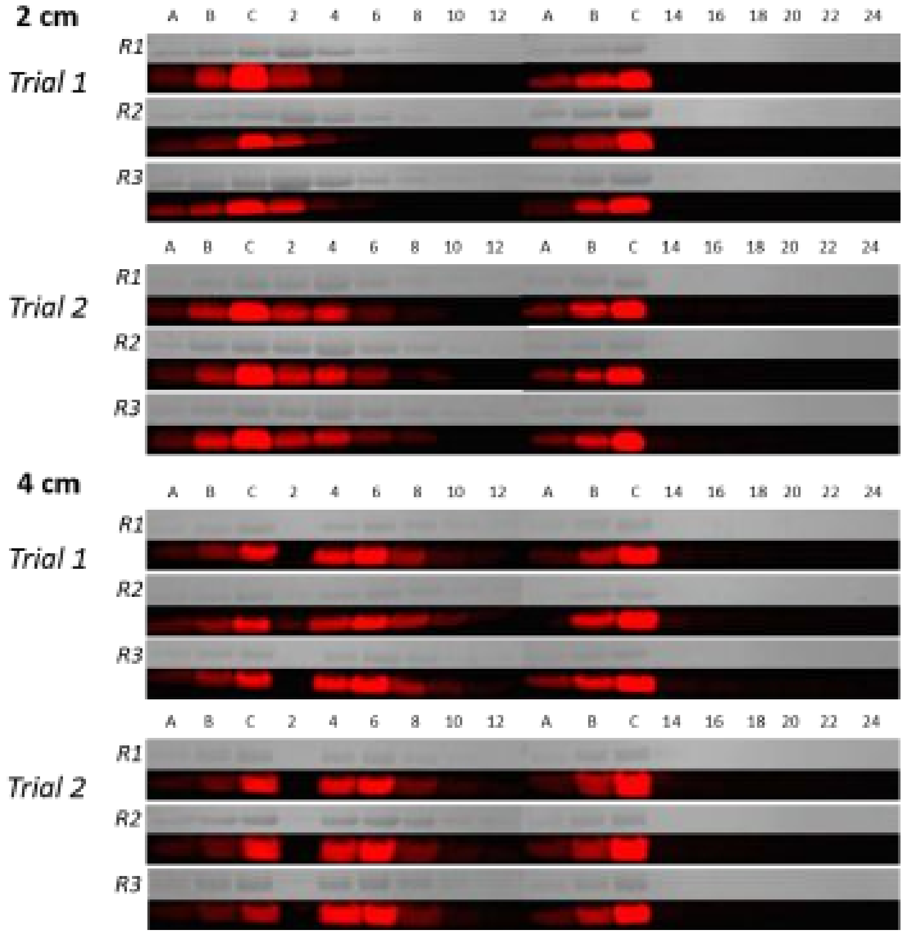
PhMV



PhMV-Cy5

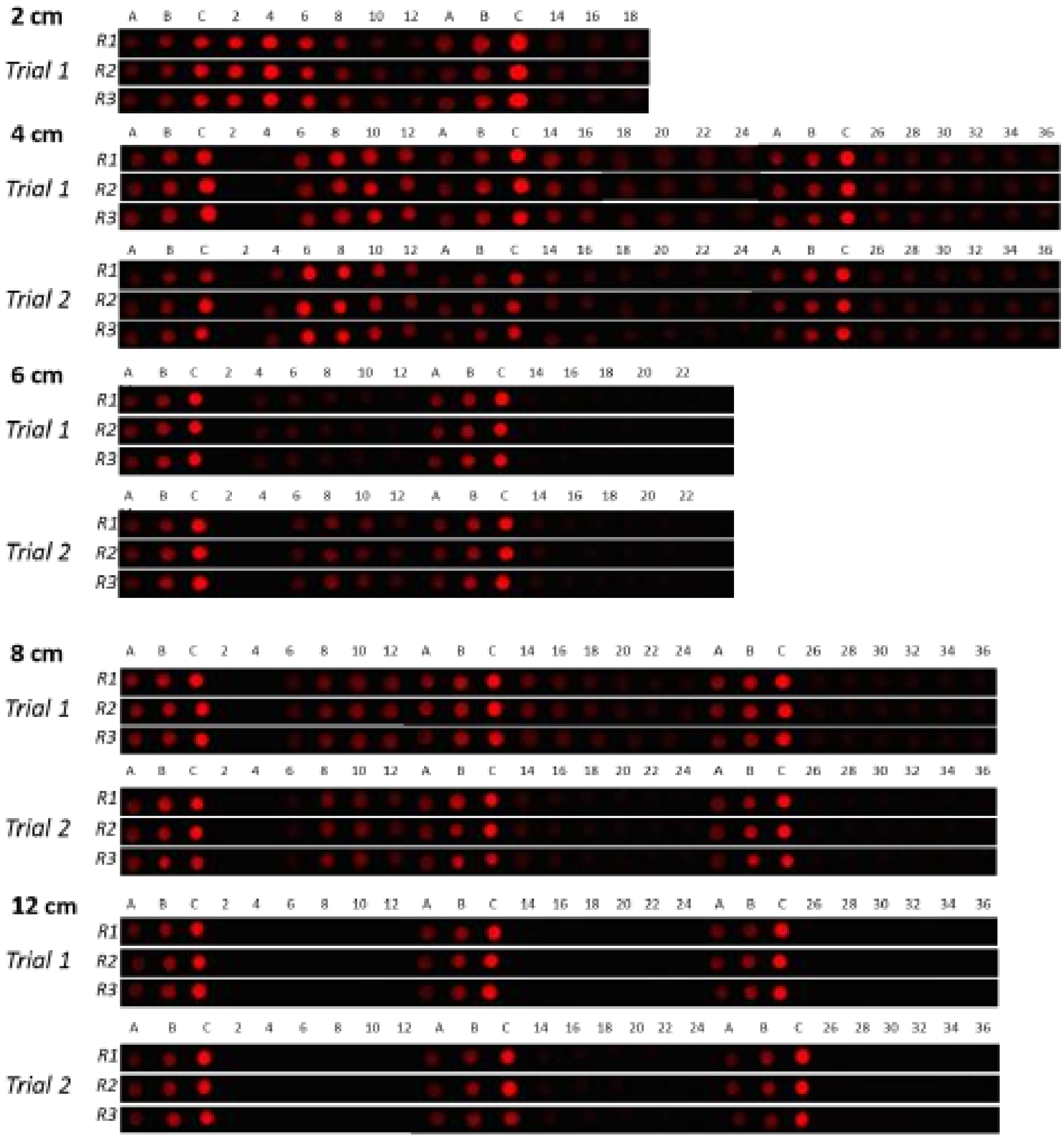


PhMV*Cy5

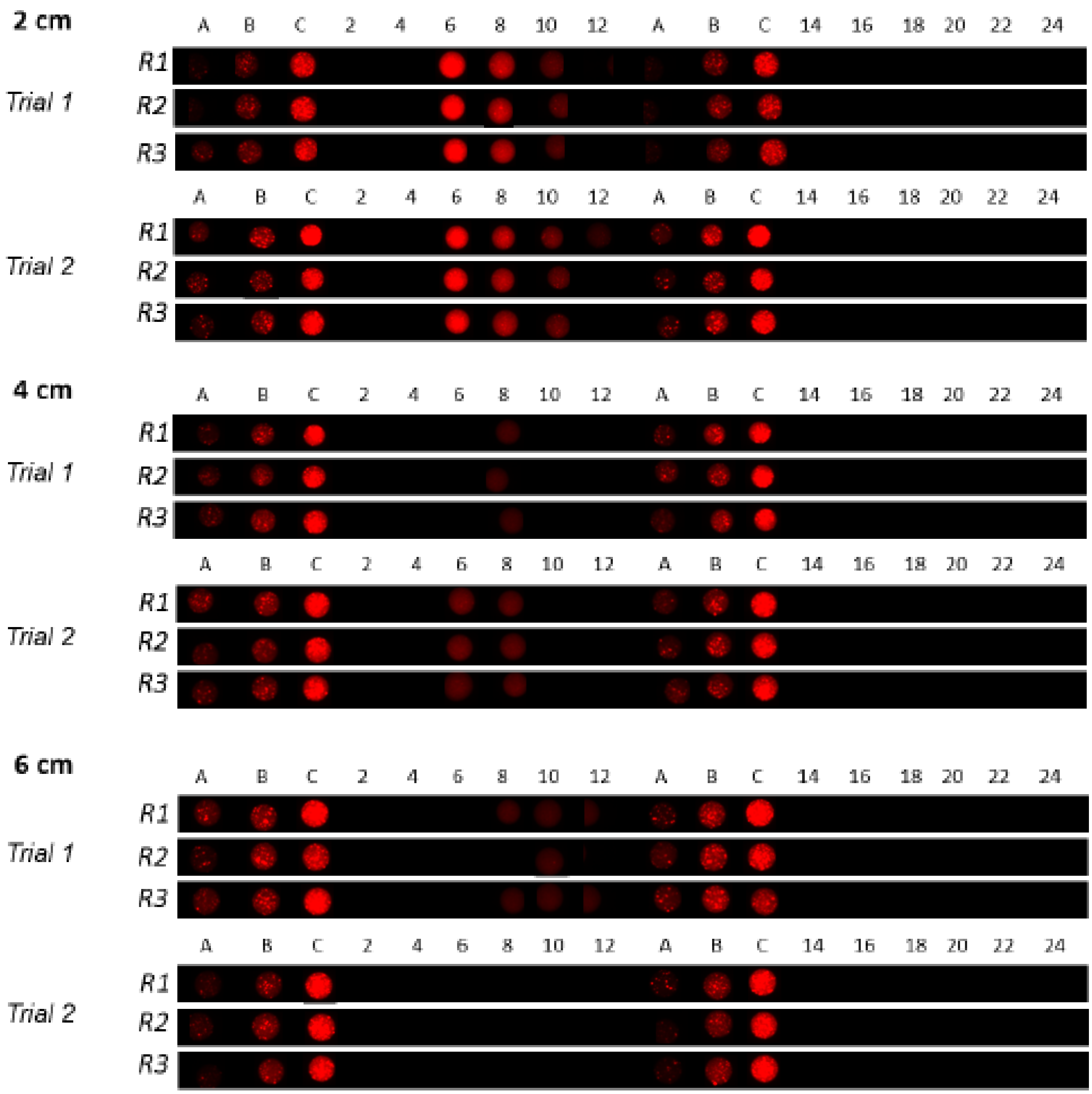


2 cm
Trial 1
Trial 2
4 cm
Trial 1
Trial 2
R1
R2
R3
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24

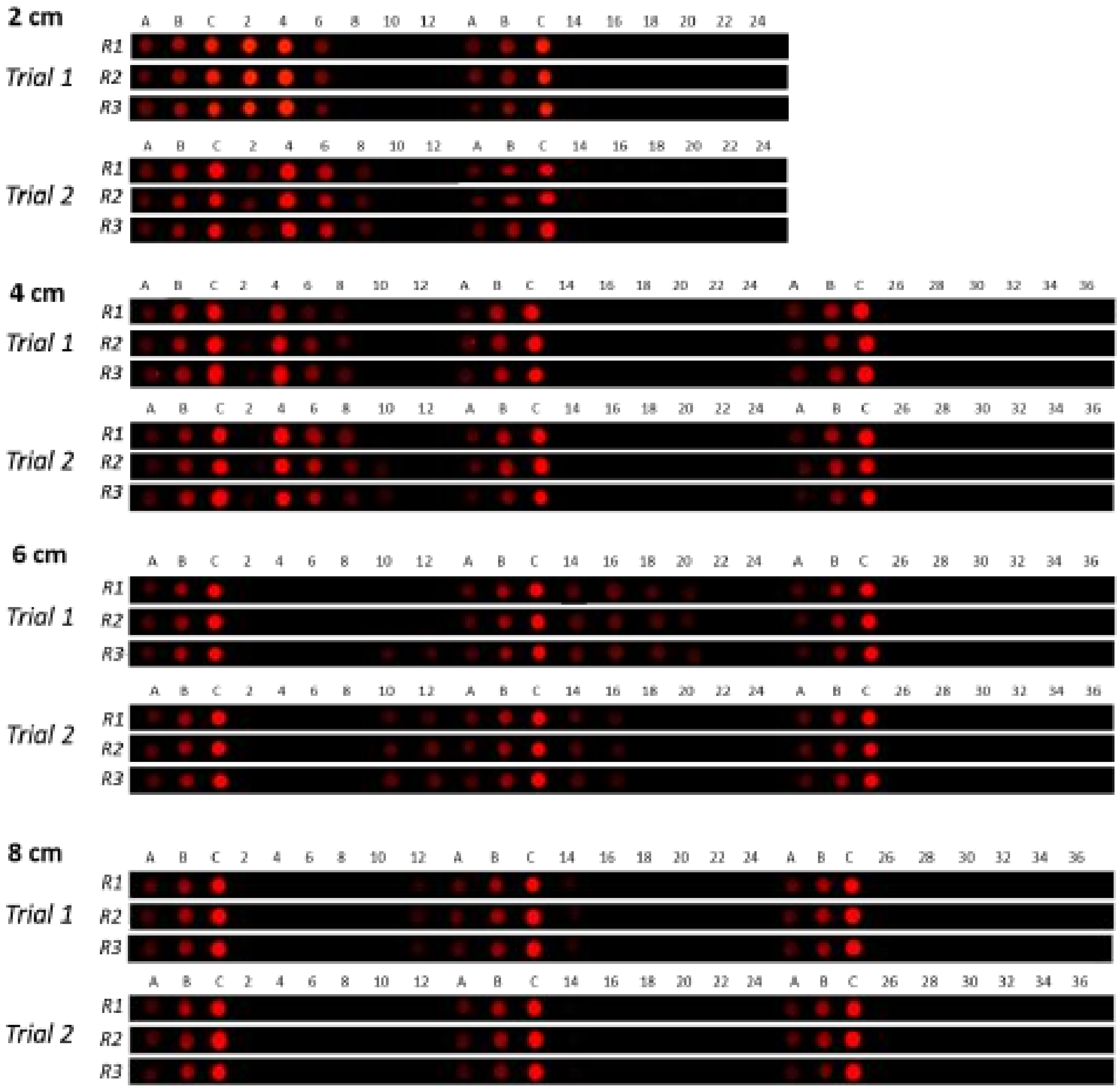
MSNP-Cy5


2 cm
Trial 1
4 cm
Trial 1
Trial 2
6 cm
Trial 1
Trial 2
8 cm
Trial 1
Trial 2
12 cm
Trial 1
Trial 2
R1
R2
R3

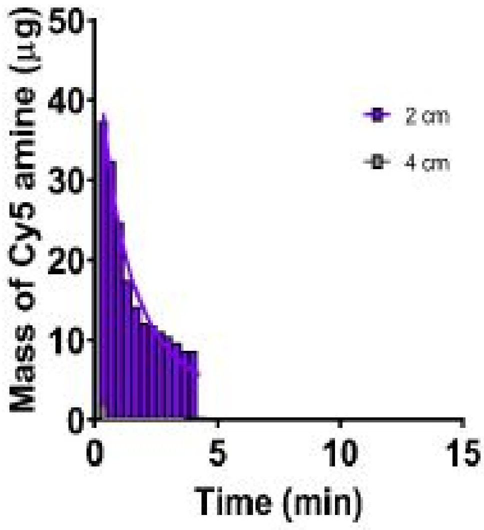
MSNP*Cy5


2 cm
4 cm
6 cm
Trial 1
Trial 2
R1
R2
R3
A B C 2 4 6 8 10 12 A B C 14 16 18 20 22 24

PLGA*Cy5



Supplementary Fig. 5 | SDS-PAGE and dot blot analysis of nanoparticle elution fractions exiting the soil column.



Supplementary Fig. 6 | Distribution of free Cy5 in the soil.

Supplementary Table 4 | Tables of virus recovery from the empty elution samples.

Elution samples lacking viral mass according to SDS-PAGE analysis were pooled and centrifuged at 112,000 g for 3 h to collect any trace amounts of virus. The concentrated solutions were analysed by SDS-PAGE gels to determine the residual mass of experimentally lost nanoparticles.

TMGMV

	Average Loss (ug)	% loss
4 cm	16	1.6
8 cm	10	1
12 cm	44	4.4
16 cm	37	3.7
20 cm	26	2.6
30 cm	28	2.8

CPMV

	Average Loss (ug)	% loss
4 cm	25	2.5
8 cm	22	2.2
12 cm	43	4.3
16 cm	32	3.2
20 cm	26	2.6
30 cm	40	4

PhMV

	Average Loss (ug)	% loss
2 cm	0	0
4 cm	0	0
6 cm	0	0
8 cm	0	0

TMGMV-Cy5

	Average Loss (ug)	% loss
4 cm	16	2.4
16 cm	37	6
30 cm	28	8.3

CPMV-Cy5

	Average Loss (ug)	% loss
4 cm	26	2.6
16 cm	31	3.1
30 cm	15	1.5

PhMV-Cy5

	Average Loss (ug)	% loss
2 cm	19	1.9
4 cm	35	3.5

TMGMV-Cy5-Amine

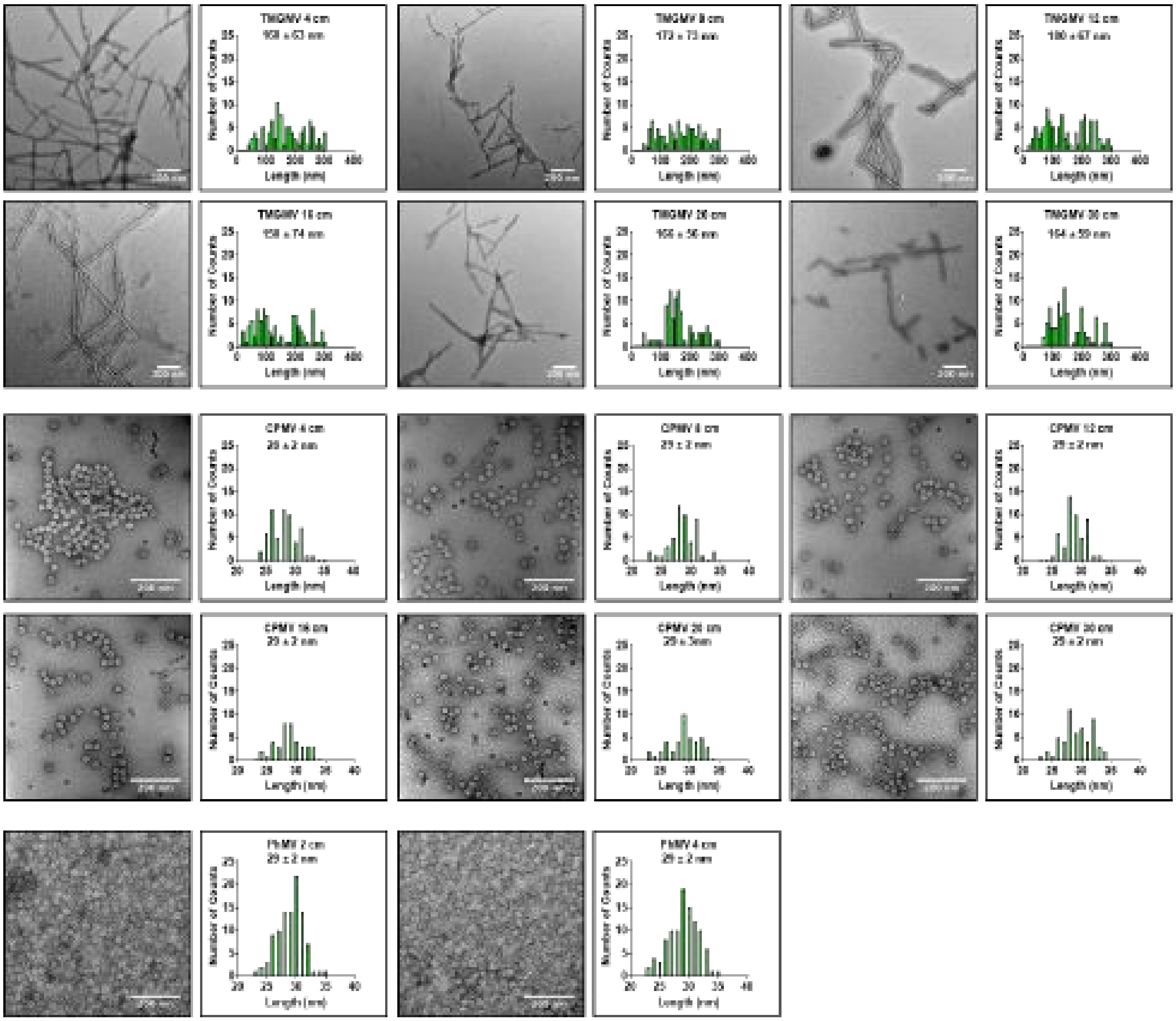
	Average Loss (ug)	% loss
2 cm	35	3.5
4 cm	20	3

CPMV-Cy5-Amine

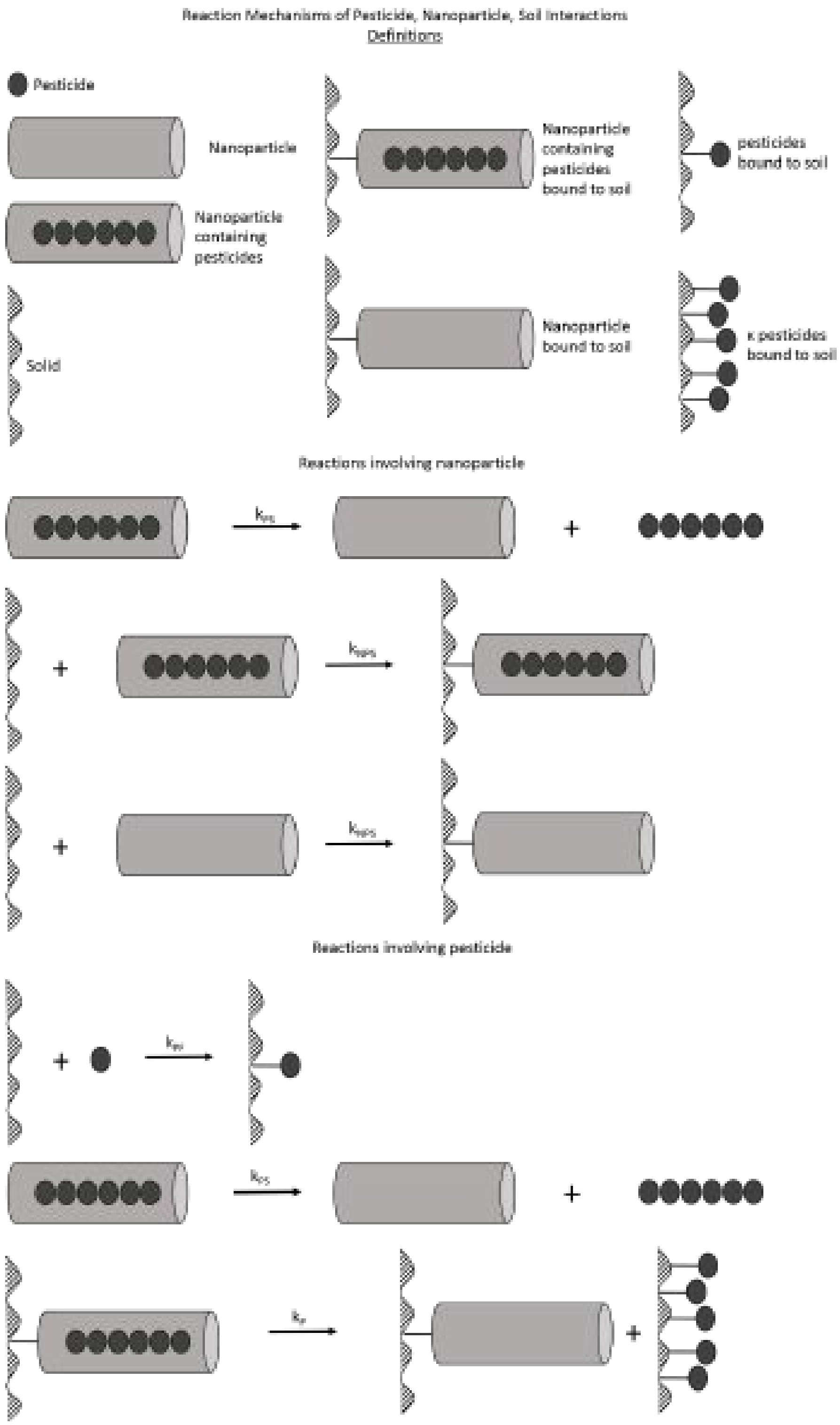
	Average Loss (ug)	% loss
4 cm	24	2.4
16 cm	16	1.6
30 cm	19	1.9

PhMV-Cy5-Amine

	Average Loss (ug)	% loss
2 cm	1.2	0.12
4 cm	1.8	0.18



Supplementary Fig. 7 | TEM images and corresponding size distribution of viruses that were leached through the soil column at different soil depths.



Supplementary Fig. 8 | Schematic of the reaction mechanisms of nanoparticles and pesticides in soil.

Supplementary Equation 1. Detailed computational methods for the numerical solution to the model of nanopesticide transport through the soil.

For the numerical solution of simultaneous partial differential equations, we used the MATLAB code "pdepe". For numerical stability, we added a "false" diffusion term for equations involving the soil particles:

$$\frac{\partial \Omega_{NPS}}{\partial t} = D_F \frac{\partial^2 \Omega_{NPS}}{\partial z^2} + \phi k_{NPS} \Omega_{NP}$$

$$\frac{\partial C_{NPS}}{\partial t} = D_F \frac{\partial^2 C_{NPS}}{\partial z^2} + \phi \left(R_P - R_{PNP} \right)$$

$$\frac{\partial C_{PS}}{\partial t} = D_F \frac{\partial^2 C_{PS}}{\partial z^2} - \phi \left(R_{PS} + R_P \right)$$

where the coefficient $\mathbf{D_F}$ **[cm^2 min^{-1}]** is given an arbitrarily small value. The boundary conditions for these modified equations are

$$z = 0: \ \Omega_{NPS} = 0; \ \ C_{NPS} = 0; \ C_{PS} = 0$$

$$z = L: \ \ \frac{\partial \Omega_{NPS}}{\partial z} = 0, \ \ \frac{\partial C_{NPS}}{\partial z} = 0, \ \ \frac{\partial C_{PS}}{\partial z} = 0$$

Dimensionless forms

We can express the nanoparticle density as a function of fluid volume within the soil column in dimensionless form by defining the dimensionless variables:

$$t' = \frac{Q}{LA\varepsilon} t \qquad z' = \frac{z}{L}$$

where **Q** is the constant volume flow of species through voids in soil, **t** is the time variable, **L** is the column length, and **A** is the constant cross-sectional area of the column.

Also, the dimensionless forms of pesticide concentration and nanoparticle density are:

$$\Omega'_{NP} = \frac{\Omega_{NP}}{\Omega^0_{NP}};\ \Omega'_{NPS} = \frac{\Omega_{NPS}}{\Omega^0_{NP}};\ C'_{NPF} = \frac{C_{NPF}}{C^0_{NPF}};\ C'_{NPS} = \frac{C_{NPS}}{C^0_{NPF}};\ C'_{P} = \frac{C_{P}}{C^0_{NPF}};\ C'_{PS} = \frac{C_{PS}}{C^0_{NPF}}$$

Starting with the equation for nanoparticle mass density in fluid:

$$\frac{\partial \Omega_{NP}}{\partial t} + \frac{Q}{A\varepsilon}\frac{\partial \Omega_{NP}}{\partial z} = D_{NP}\frac{\partial^2 \Omega_{NP}}{\partial z^2} + \left(\frac{1-\varepsilon}{\varepsilon}\right)\phi R_{NPS}, \quad 0<z<L$$

we substitute the dimensionless variables:

$$\frac{Q\Omega^0_{NP}}{LA\varepsilon}\frac{\partial \Omega'_{NP}}{\partial t'} + \frac{Q\Omega^0_{NP}}{LA\varepsilon}\frac{\partial \Omega'_{NP}}{\partial z'} = \frac{\Omega^0_{NP}D_{NP}}{L^2}\frac{\partial^2 \Omega'_{NP}}{\partial z'^2} - \left(\frac{1-\varepsilon}{\varepsilon}\right)\phi\Omega^0_{NP}k_{NPS}\Omega'_{NP}, \quad 0<z'<1$$

Dividing by the coefficient of the first term yields

$$\frac{\partial \Omega'_{NP}}{\partial t'} + \frac{\partial \Omega'_{NP}}{\partial z'} = \left[\frac{A\varepsilon D_{NP}}{QL}\right]\frac{\partial^2 \Omega'_{NP}}{\partial z'^2} - \left[\frac{LA\varepsilon\phi k_{NPS}}{Q}\right]\left[\frac{1-\varepsilon}{\varepsilon}\right]\Omega'_{NP}, \quad 0<z'<1 \qquad \textbf{(1)}$$

Starting with the equation for mass density of nanoparticles attached to soil particles:

$$\frac{\partial \Omega_{NPS}}{\partial t} = D_F\frac{\partial^2 \Omega_{NPS}}{\partial z^2} + \phi k_{NPS}\Omega_{NP}, \quad 0<z<L$$

we substitute the dimensionless variables:

$$\frac{Q\Omega^0_{NP}}{LA\varepsilon}\frac{\partial \Omega'_{NPS}}{\partial t'} = \frac{D_F\Omega^0_{NP}}{L^2}\frac{\partial^2 \Omega'_{NPS}}{\partial z^2} + \phi k_{NPS}\Omega^0_{NP}\Omega'_{NP}, \quad 0<z'<1$$

Dividing by the coefficient of the first term yields:

$$\frac{\partial \Omega'_{NPS}}{\partial t'} = \left[\frac{D_F A\varepsilon}{QL}\right]\frac{\partial^2 \Omega'_{NPS}}{\partial z'^2} + \left[\frac{LA\varepsilon\phi k_{NPS}}{Q}\right]\Omega'_{NPS}, \quad 0<z'<1 \qquad \textbf{(2)}$$

The equation for pesticide dissolved in fluid can be written as:

$$\frac{\partial C_P}{\partial t} + \frac{Q}{A\varepsilon}\frac{\partial C_P}{\partial z} = D_P\frac{\partial^2 C_P}{\partial z^2} - R_{PF} + \left(\frac{1-\varepsilon}{\varepsilon}\right)\phi R_{PS}, \quad 0<z<L$$

After the substitution of dimensionless variables, we find:

$$\frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_P}{\partial t'} + \frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_P}{\partial z'} = \frac{D_P C^0_{NPF}}{L^2}\frac{\partial^2 C'_P}{\partial z'^2} + k_{PF}C^0_{NPF}C'_{NPF} - \left(\frac{1-\varepsilon}{\varepsilon}\right)\phi\left[k_{PS}C^0_{NPF}\right]C'_P \quad 0<z'<1$$

Dividing by the coefficient of the first term yields:

$$\frac{\partial C'_P}{\partial t'}+\frac{\partial C'_P}{\partial z'}=\left[\frac{A\varepsilon D_P}{QL}\right]\frac{\partial^2 C'_P}{\partial z'^2}+\left[\frac{LA\varepsilon k_{PF}}{Q}\right]C'_{NPF}-\left[\frac{LA\varepsilon\phi k_{PS}}{Q}\right]\left[\frac{1-\varepsilon}{\varepsilon}\right]C'_P, \qquad 0<z'<1 \qquad \textbf{(3)}$$

The equation for pesticide attached to nanoparticles in fluid can be written as:

$$\frac{\partial C_{NPF}}{\partial t}+\frac{Q}{A\varepsilon}\frac{\partial C_{NPF}}{\partial z}=D_{NP}\frac{\partial^2 C_{NPF}}{\partial z^2}+R_{PF}+\left(\frac{1-\varepsilon}{\varepsilon}\right)\phi R_{PNP}, \qquad 0<z<L$$

After the substitution of dimensionless variables, we find:

$$\frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_{NPF}}{\partial t'}+\frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_{NPF}}{\partial z'}=\frac{D_{NP}C^0_{NPF}}{L^2}\frac{\partial^2 C'_{NPF}}{\partial z'^2}-k_{PF}C^0_{NPF}C'_{NPF}-\left(\frac{1-\varepsilon}{\varepsilon}\right)\phi k_{NPS}C^0_{NPF}C'_{NPF}$$

Dividing by the coefficient of the first term yields:

$$\frac{\partial C'_{NPF}}{\partial t'}+\frac{\partial C'_{NPF}}{\partial z'}=\left[\frac{A\varepsilon D_{NP}}{QL}\right]\frac{\partial^2 C'_{NPF}}{\partial z'^2}-\left[\frac{LA\varepsilon k_{PF}}{Q}\right]C'_{NPF}+\left[\frac{LA\varepsilon\phi k_{NPS}}{Q}\right]\left(\frac{1-\varepsilon}{\varepsilon}\right)C'_{NPF}, \ 0<z'<1 \qquad \textbf{(4)}$$

The equation for pesticide attached to nanoparticles on soil particles:

$$\frac{\partial C_{NPS}}{\partial t}=D_F\frac{\partial^2 C_{NPS}}{\partial z^2}+\phi\left(R_P-R_{PNP}\right)$$

After the substitution of dimensionless variables, we find:

$$\frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_{NPS}}{\partial t'}=\frac{D_F C^0_{NPF}}{L^2}\frac{\partial^2 C'_{NPS}}{\partial z'^2}-\phi\left(k_P C^0_{NPF}C'_{NPS}-k_{NPS}C^0_{NPF}C'_{NPF}\right)$$

Dividing by the coefficient of the first term yields:

$$\frac{\partial C'_{NPS}}{\partial t'}=\left[\frac{D_F A\varepsilon}{QL}\right]\frac{\partial^2 C'_{NPS}}{\partial z'^2}-\left[\frac{LA\varepsilon\phi k_P}{Q}\right]C'_{NPS}+\left[\frac{LA\varepsilon\phi k_{NPS}}{Q}\right]C'_{NPF} \qquad \textbf{(5)}$$

The equation for pesticide adsorbed by soil particles can be written as:

$$\frac{\partial C_{PS}}{\partial t}=D_F\frac{\partial^2 C_{PS}}{\partial z^2}-\phi\left(R_{PS}+R_P\right)$$

After the substitution of dimensionless variables, we find:

$$\frac{QC^0_{NPF}}{LA\varepsilon}\frac{\partial C'_{PS}}{\partial t'}=\frac{D_F C^0_{NPF}}{L^2}\frac{\partial^2 C'_{PS}}{\partial z^2}+\phi\left(k_{PS}C^0_{NPF}C'_P+k_P C^0_{NPF}C'_{NPS}\right)$$

Dividing by the coefficient of the first term yields:

$$\frac{\partial C'_{PS}}{\partial t'}=\left[\frac{D_F A\varepsilon}{QL}\right]\frac{\partial^2 C'_{PS}}{\partial z^2}+\left[\frac{LA\varepsilon\phi k_{PS}}{Q}\right]C'_P+\left[\frac{LA\varepsilon\phi k_P}{Q}\right]C'_{NPS} \qquad \textbf{(6)}$$

Equations 1-6 can be written as:

$$\frac{\partial \Omega'_{NP}}{\partial t'} + \frac{\partial \Omega'_{NP}}{\partial z'} = \frac{1}{Pe_{NP}} \frac{\partial^2 \Omega'_{NP}}{\partial z'^{\,2}} - Da_{NP}\left[\frac{1-\varepsilon}{\varepsilon}\right]\Omega'_{NP} \tag{1}$$

$$\frac{\partial \Omega'_{NPS}}{\partial t'} = \frac{1}{Pe_{NPS}} \frac{\partial^2 \Omega'_{NPS}}{\partial z'^2} + Da_{NP}\Omega'_{NP} \tag{2}$$

$$\frac{\partial C'_{P}}{\partial t'} + \frac{\partial C'_{P}}{\partial z'} = \frac{1}{Pe_{D}} \frac{\partial^2 C'_{P}}{\partial z'^2} + Da_{PF}C'_{NPF} - Da_{PS}\left[\frac{1-\varepsilon}{\varepsilon}\right]C'_{P} \tag{3}$$

$$\frac{\partial C'_{NPF}}{\partial t'} + \frac{\partial C'_{NPF}}{\partial z'} = \frac{1}{Pe_{NP}} \frac{\partial^2 C'_{NPF}}{\partial z'^2} - Da_{PF}C'_{NPF} + Da_{NP}\left(\frac{1-\varepsilon}{\varepsilon}\right)C'_{NPF} \tag{4}$$

$$\frac{\partial C'_{NPS}}{\partial t'} = \frac{1}{Pe_{NPS}} \frac{\partial^2 C'_{NPS}}{\partial z'^2} - Da_{P}C'_{NPS} + Da_{NP}C'_{NPF} \tag{5}$$

$$\frac{\partial C'_{PS}}{\partial t'} = \frac{1}{Pe_{NPS}} \frac{\partial^2 C'_{PS}}{\partial z^2} + Da_{PS}C'_{P} + Da_{P}C'_{NPS} \tag{6}$$

Supplementary Code 1. MATLAB code.

Part 1: Model for nanoparticle transport through soil

```
% Last edited: 07/21/18

function NanoparticlesPdepe
tic
clc
clear
%% Defining the constants
E = 0.45; % Volume fraction of fluid [1 = 100% fluid]
rsf = (1-E)/E; % Particle-to-fluid volume ratio [dimensionless]
Q = 1.5; % Flow rate [cm3/min]
Phi = 3401; % Adsorption surface per particle volume [cm^-1]
radius = 1.4; % Radius of the soil column [cm]
A = pi*radius^2; % Cross sectional Area of the column [cm^2]
massparticle = 1; % Mass of particle injected in the system [mg]
volumeinj = 0.3; % Volume of particle injection [cm^3]
OmegaVfinitial = massparticle/volumeinj; % Initial nanoparticle concentration [mg/cm^3]

%% Experimental data
ExpData = xlsread('ExperimentalDataMatlab_TMGMVnodelay'); % Importing the experimental data
% ~~First row is the depth value~~
Depth = ExpData (1,:,:); % Creating the array of soil depth data
Time = ExpData (2:end,1,:); % Creating the array of time data.
MassNP = ExpData (2:end,2:end,:); % Creating the array of experimental mass of nanoparticles
ConcNP = MassNP /500; % Converting the mass into concentration of nanoparticles

%% Initialization of parameters
Cerror = zeros (length (Depth)-1);
individualparams= zeros (length (Depth)-1, 2); % We currently only have a Peclet number and a Damkohler
number as parameters (2 parameters)
global optimj
global L

%% Parameters estimation STARTS
for optimj = 1: length(Depth)-1
    %% Non dimensionalization of the time and length
    L = Depth(optimj+1);
```

```
KT= Q/(L*A*E); % Dimensionless constant of time
tinj=volumeinj/Q/2*KT; % Initial time of particle injection into the system
TimeD = Time * KT; % Dimensionless time
depthD = Depth/L; % Dimensionless depth

%% Defining dimensionless time and length for the PDEPE function
m = 0; % The symmetry of the problem. m can be slab = 0, cylindrical = 1, or spherical = 2.
x = linspace(0,1,1000); % Dimensionless
tmin = 0;
tmax = ceil(max(TimeD));
t = linspace(tmin,tmax,1001); % Dimensionless

%% Experimental data as a function of t for specific x values
ConcNPD = ConcNP./OmegaVfinitial; % Dimensionless concentration
  fitobject = fit(TimeD,ConcNPD(:,optimj),'smoothingspline'); % Creating a curve fit to the
  experimental data
FittedConc = fitobject (t); % Extracting more points from the curve fit of the experimental data
for i = 1: length (Time)-1
    OmegaNPF_exp(i+1,:)=ConcNPD(i+1,:); % The first index is the depth row
end

%% Parameter estimation function
  options1 = optimoptions('fmincon','Display','iter','Algorithm', 'sqp') % Options for the fmincon
  function
[optparams Error2] = fmincon(@objfun, [0.2 2],[],[],[],[],[0.05 0],[5 15],[], options1) % Estimating
the value of the Peclet number and a Damkohler number of the nanoparticle
individualparams(optimj,:) = optparams % Returns the optimized parameter for each depth
  Dv (optimj) =  (individualparams(optimj,1)*Q*L)/(A*E) % Computing the value of the dispersion
  coefficient of the nanoparticle
kvfp (optimj) =  (individualparams(optimj,2)*Q)/(L*A*E*Phi) % Computing the value of the rate of
binding to soil of the nanoparticle
ERROR (optimj) = Error2 % Returns the absolute error between the model and the experimental data for
each depth (~~Scaled by 1000X~~)
Averageparameters = mean (individualparams); % Returns the average value of the Peclet number and
Damkohler number
Standardparameters = std (individualparams); % Returns the standard derivation value of the Peclet
number and Damkohler number

%% Figures of the results
  maxth = max(OmegaVfinitial.*OmegaNPF_x(:,1)); maxexp = max(smooth(ConcNP(:,1))); % The highest peak
  is the first depth
```

```
MaxY = max ([maxexp maxth]); % To normalize graph with dimension to the highest peak

maxthD = max(OmegaNPF_x(:,1)); maxexpD = max(smooth(ConcNPD(:,1)));
MaxYD = max ([maxexpD maxthD]); % To normalize dimensionless graphs to the highest peak

figure(1) % Nanoparticles transport in fluid
bx1 = subplot (2,1,1);
plot (TimeD/KT, (ConcNP(:,optimj)),'Color',[0.15*optimj, 0,1])
hold on
xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Experimental Data'); axis tight;

bx2 = subplot (2,1,2);
plot(t/KT,OmegaVfinitial*OmegaNPF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
hold on
xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Model Output'); axis tight
bx1.YLim = [0,MaxY]; bx2.YLim = [0,MaxY];
bx2.XLim = [0,max(Time)];

%% Exporting results in an Excel sheet
EXP = [Time, ConcNP]';
Time_m = (t/KT)';
Model(optimj,1:length(Time_m))= Time_m;
Model(length(Depth)-1+optimj ,1:length(OmegaNPF_x))= OmegaVfinitial*OmegaNPF_x(:,optimj);
xlswrite('Particlemodel.xlsx',Model'); % Model output
xlswrite('Particleexp.xlsx',EXP');% Experimental Ddta

  end

function Error = objfun (params)
    %% UNKNOWN parameter to fit
    Pev_inv = params(1); % a Peclet number of nanoparticles
    Dav = params(2); % a Damkohler number of nanoparticles

    %% Solving the PDEPE
    % options2=odeset('RelTol',1e-5); % sets the relative tolerance to 10^-5
    sol = pdepe(m,@pdeNPSoil,@pdeNPSoilic,@pdeNPSoilbc,x,t);
    OmegaNPF = sol(:,:,1); % Model output of nanoparticles in fluid in time and space

    % OmegaNPF as a function of t at the exit of the column
    OmegaNPF_x(:,optimj)=OmegaNPF(:,length(x));
```

```
        %% Comparing concentrations and calculating error for specific time points
        Cerror(optimj)=0;
        for xi = 1 : length (t)
           Cerror(optimj) = Cerror(optimj)+ (OmegaNPF_x(xi,optimj)-FittedConc(xi,:))^2;
        end
        Error=Cerror(optimj)*1000

        %% Inner Functions
        function [c,f,s] = pdeNPSoil(x,t,u,DuDx)
            % Defining c, f, and s to solve the PDEPE function
            c = [1];  % Coefficients in front of DuDt term (none here so all ones)
            f = [Pev_inv*DuDx(1)];  % Coefficients for the second derivative
            s = [-DuDx(1) - Dav*rsf*u(1)]; % Reaction term
        end

        %% Initial Conditions
        function u0 = pdeNPSoilic(x)
            u0 = [0];
        end
        % Boundary conditions
        function [pl,ql,pr,qr] = pdeNPSoilbc(xl,ul,xr,ur,t)
            OmegaVfi = 1- 1/(1+exp(-50*(t-tinj)));
            pl = [ul(1)-OmegaVfi];
            ql = [0];
            pr = [0];
            qr = [1];
        end
    end

%%
WarnWave = [sin(1:.6:400), sin(1:.7:400), sin(1:.4:400)]; % Produces a sound alert when code is done
running
Audio = audioplayer(WarnWave, 22050);
play(Audio);
toc

end
```

Part 2: Model for pesticides transport through soil

```
% Last edited: 07/21/18

function DrugonlyPdepe1
tic
clc
clear
%% Defining the constants
E = 0.45; % Volume fraction of fluid  [1 = 100% fluid]
rsf = (1-E)/E; % Particle-to-fluid volume ratio [dimensionless]
Q = 1.5; % Flow rate [cm3/min]
Phi = 3401; % Adsorption surface per particle volume[cm^-1]
radius = 0.5; % Radius of the soil column [cm]
A = pi*radius^2; % Cross sectional Area of the column[cm^2]
massparticle = 0.5; % Mass of particle injected in the system [mg]
volumeinj = 0.3; % Volume of particle injection [cm^3]
OmegaVfinitial = massparticle/volumeinj; % [mg/cm^3]

%% Experimental Data
ExpData = xlsread('ExperimentalDataMatlab_DRUGnodelay'); % Importing the experimental data
% ~~First row is the depth value~~
Depth = ExpData (1,:,:); % Creating the array of soil depth data
Time = ExpData (2:end,1,:); % Creating the array of time data.
MassNP = ExpData (2:end,2:end,:); % Creating the array of experimental mass of pesticides
ConcNP = MassNP /500; % converting the mass into concentration of pesticides

%% Initialization of parameters
Cerror = zeros (length (Depth)-1);
individualparams= zeros (length (Depth)-1, 2); % We currently only have a Peclet number and a Damkohler
number as parameters (2 parameters)
global optimj
global L

%% Parameters estimation STARTS
for optimj = 1: length(Depth)-1
    %% Non dimensionalization of the time and length
    L = Depth(optimj+1);
    KT= Q/(L*A*E); % Dimensionless constant of time
    tinj=volumeinj/Q/2*KT; % Initial time of particle injection into the system
```

```
TimeD = Time * KT; % Dimensionless time
depthD = Depth/L; % Dimensionless depth

%% Defining dimensionless time and length for the PDEPE function
m = 0; % The symmetry of the problem. m can be slab = 0, cylindrical = 1, or spherical = 2.
x = linspace(0,1,1000); % Dimensionless
tmin = 0;
tmax = ceil(max(TimeD));
t = linspace(tmin,tmax,1001); % Dimensionless

%% Experimental data as a function of t for specific x values
ConcNPD = ConcNP./OmegaVfinitial; % Dimensionless concentration
  fitobject = fit(TimeD,ConcNPD(:,optimj),'smoothingspline');% Creating a curve fit to the experimental
  data
FittedConc = fitobject (t); % Extracting more points from the curve fit of the experimental data
for i = 1: length (Time)-1
    OmegaNPF_exp(i+1,:)=ConcNPD(i+1,:); % The first index is depth row
end

%% Parameter estimation function
  options1 = optimoptions('fmincon','Display','iter','Algorithm', 'sqp') % Options for the fmincon
  function
[optparams Error2] = fmincon(@objfun, [0.05 0.01],[],[],[],[],[0 0.001],[5 15],[], options1) %
Estimating the value of the Peclet number and a Damkohler number of the nanoparticle
individualparams(optimj,:) = optparams % Returns the optimized parameter for each depth
  ERROR (optimj) = Error2 % Returns the absolute error between the model and the experimental data for
  each depth (~~Scaled by 1000X~~)

%% Figures of the results
  maxth = max(OmegaVfinitial.*OmegaNPF_x(:,1)); maxexp = max(smooth(ConcNP(:,1))); % The highest peak
  is the firts depth
MaxY = max ([maxexp maxth]); % To normalize graph with dimension to the highest peak

maxthD = max(OmegaNPF_x(:,1)); maxexpD = max(smooth(ConcNPD(:,1)));
MaxYD = max ([maxexpD maxthD]); % To normalize dimensionless graphs to the highest peak

figure(1) % Pesticide transport in fluid
bx1 = subplot (2,1,1);
plot (TimeD/KT, (ConcNP(:,optimj)),'Color',[0.15*optimj, 0,1])
hold on
xlabel('Time [min]'); ylabel('\Omega_D_F [mg/cm^3]');title ('Experimental Data'); axis tight;
```

```
bx2 = subplot (2,1,2);
plot(t/KT,OmegaVfinitial*OmegaNPF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
hold on
xlabel('Time [min]'); ylabel('\Omega_D_F [mg/cm^3]');title ('Model Output'); axis tight
bx1.YLim = [0,MaxY];bx2.XLim = [0,max(Time)];
bx2.YLim = [0,MaxY];bx2.XLim = [0,max(Time)];

%% Exporting results in an Excel sheet
EXP = [Time, ConcNP]';
Time_m = (t/KT)';
Model(optimj,1:length(Time_m))= Time_m;
Model(length(Depth)-1+optimj ,1:length(OmegaNPF_x))= OmegaVfinitial*OmegaNPF_x(:,optimj);
xlswrite('DRUGmodel.xlsx',Model');
xlswrite('DRUGexp.xlsx',EXP');

  end

function Error = objfun (params)
    %% UNKNOWN parameter to fit
    Ped_inv = params(1); % a Peclet number of pesticides
    Dad = params(2); % a Damkohler number of pesticides

    %% Solving the PDEPE
    % options2=odeset('RelTol',1e-5); % sets the relative tolerance to 10^-5
    sol = pdepe(m,@pdeNPSoil,@pdeNPSoilic,@pdeNPSoilbc,x,t);
    OmegaNPF = sol(:,:,1); % Model output of pesticides in fluid in time and space

    % OmegaDF as a function of t at the exit of the column
    OmegaNPF_x(:,optimj)=OmegaNPF(:,length(x));

    %% Comparing concentrations and calculating error for specific time points
    Cerror(optimj)=0;
    for xi = 1 : length (t)
        Cerror(optimj) = Cerror(optimj)+ (OmegaNPF_x(xi,optimj)-FittedConc(xi,:))^2;
    end
    Error=Cerror(optimj)*1000

    %% Inner Functions
    function [c,f,s] = pdeNPSoil(x,t,u,DuDx)
        % Defining c, f, and s to solve the PDEPE function
```

```
            c = [1];  % Coefficients in front of DuDt term (none here so all ones)
            f = [Ped_inv*DuDx(1)];  % Coefficients for the second derivative
            s = [-DuDx(1) - Dad*rsf*u(1)]; % Reaction term
        end

        %% Initial Conditions
        function u0 = pdeNPSoilic(x)
            u0 = [0];
        end
        % Boundary conditions
        function [pl,ql,pr,qr] = pdeNPSoilbc(xl,ul,xr,ur,t)
            OmegaVfi = 1- 1/(1+exp(-50*(t-tinj)));
            pl = [ul(1)-OmegaVfi];
            ql = [0];
            pr = [0];
            qr = [1];
        end
    end
%%
WarnWave = [sin(1:.6:400), sin(1:.7:400), sin(1:.4:400)]; % Produces a sound alert when code is done
running
Audio = audioplayer(WarnWave, 22050);
play(Audio);
toc
%%
end
```

Part 3: Model of pesticide release from nanoparticles

```
% Last edited: 07/23/18

function DruginNanoparticlesPdepe
tic
clc
clear
%% Defining the constants
E = 0.45; % Volume fraction of fluid  [1 = 100% fluid]
rsf = (1-E)/E; % Particle-to-fluid volume ratio [dimensionless]
Q = 1.5; % Flow rate [cm3/min]
Phi = 3401; % Adsorption surface per particle volume[cm^-1]
radius = 1.4; % Radius of the soil column [cm]
A = pi*radius^2; % Cross sectional Area of the column[cm^2]
massparticle = 1; % Mass of particle injected in the system [mg]
MWDrug = 653.77; % Molecular weight of pesticide (g/mol)
%moleDrug = 2.67857142e-009 ; % (CPMV) moles of pesticide added to the system (mol)
%moleDrug = 5.32995e-009 ; % (TMGMV) moles of pesticide added to the system (mol)
%moleDrug = 1.17021e-008 ; % (PHMV) moles of pesticide added to the system (mol)
moleDrug = 4.77922e-009 ; % (MSNP) moles of pesticide added to the system (mol)
massDrug = MWDrug * moleDrug * 1000;  % Mass of pesticide injected in the system [mg]
volumeinj = 0.3; % Volume of particle injection [cm^3]
OmegaVfinitial = massparticle/volumeinj; % [mg/cm^3]
OmegaDfinitial1 = massDrug/volumeinj; % [mg/cm^3]
OmegaDfinitial = massDrug/volumeinj; % [mg/cm^3]


%% Experimental Data
ExpData = xlsread('ExperimentalDataMatlab_MSNP2nodelay');% Importing the experimental data of nanoparticles
through soil
ExpData_D = xlsread('ExperimentalDataMatlab_DRUGinMSNPnodelay');% Importing the experimental data of
pesticide in nanoparticles through soil
NPParamData = xlsread('MSNPparameters'); % Importing the parameter optimization from nanoparticles
DrugParamData = xlsread('DRUGparameters2'); % Importing the parameter optimization from pesticide

%first row is the depth value
Depth = ExpData_D (1,:,:); % Creating the array of soil depth data
NPDepth = ExpData (1,:,:); % Creating the array of soil depth data
Time = ExpData_D (2:end,1,:); % Creating the array of time data.
```

```
MassNP = ExpData (2:end,2:end,:); % Creating the array of experimental mass of nanoparticles
ConcNP = MassNP /500; % Converting the mass into concentration of nanoparticles
MassD = ExpData_D (2:end,2:end,:)*1000; % Creating the array of experimental mass of pesticide
ConcD = MassD /0.5; % Converting the mass into concentration of pesticide

% Creating the vectors of Peclet numbers and  for each depth
Pev_inv = NPParamData (:,1); % Peclet number of nanoparticles
Dav = NPParamData (:,2); % a Damkohler number
Ped_inv1 = DrugParamData (:,1);% Peclet number of pesticide
Dad1 = DrugParamData(:,2)*(A/0.7854); % a Damkohler number
kd = (Dad1*Q)/(2*A*E*Phi)
Dd = (Q*2*Ped_inv1)/(A*E)
%% Initialization of parameters
Cerror = zeros (length (Depth)-1);
individualparams= zeros (length (Depth)-1, 1); % We currently only pesticide rate of release from
nanoparticles as parameters (1 parameter)
global optimj
global L

%% Parameters Estimation STARTS
for optimj = 1: length(Depth)-1
    %% Non dimensionalization of the time and length
    L = Depth(optimj+1);
    Dad = (kd*L*A*E*Phi)/Q
    Ped_inv = (Dd*A*E)/(Q*L)
    KT= Q/(L*A*E); % Dimensionless constant of time
    tinj=volumeinj/Q/2*KT; % Initial time of particle injection into the system
    TimeD = Time * KT; % Dimensionless time
    depthD = Depth/L; % Dimensionless depth

    %% Defining dimensionless time and length for the PDEPE function
    m = 0; % The symmetry of the problem. m can be slab = 0, cylindrical = 1, or spherical = 2.
    x = linspace(0,1,1000); % Dimensionless
    tmin = 0;
    tmax = ceil(max(TimeD));
    t = linspace(tmin,tmax,1001); % Dimensionless


    %% Experimental data as a function of t for specific x values
    ConcNPD = ConcNP./OmegaVfinitial; % Dimensionless concentration of nanoparticles
    ConcDD = ConcD./OmegaDfinitial; % Dimensionless concentration of nanoparticles
```

```
  fitobject = fit(TimeD,ConcNPD(:,optimj),'smoothingspline'); % Creating a curve fit to the
  experimental data of nanoparticles
FittedConc = fitobject (t); % Extracting more points from the curve fit of the experimental data of
pesticide
fitDRUG = fit(TimeD,ConcDD(:,optimj),'smoothingspline'); % Creating a curve fit to the experimental
data
FittedConcD = fitDRUG (t); % Extracting more points from the curve fit of the experimental data of
pesticide

for i = 1: length (Time)-1
    OmegaNPF_exp(i+1,:)=ConcNPD(i+1,:); % The first index is depth row (experimental data of
  nanoparticles)
end

for i = 1: length (Time)-1
    OmegaDF_exp(i+1,:)=ConcDD(i+1,:); % The first index is depth row (experimental data of pesticide)
end

%% Parameter estimation function
  options1 = optimoptions('fmincon','Display','iter','Algorithm', 'sqp') % Options for the fmincon
  function
[optparams Error2] = fmincon(@objfun, [20],[],[],[],[],[0],[100],[], options1)
individualparams(optimj,:) = optparams % returns the optimized parameter for each depth
  ERROR (optimj) = Error2 % returns the absolute error between the model and the experimental data for
  each depth

%% Figures of the results
  maxth = max(OmegaVfinitial.*OmegaNPF_x(:,1)); maxexp = max(smooth(ConcNP(:,1))); maxthDF =
  max(OmegaDfinitial.*OmegaDF_x(:,1));
maxDexp = max(smooth(ConcD(:,1))); maxthDNPF = max(OmegaDfinitial.*OmegaDNPF_x(:,1));% the highest of
nanoparticles peak is the firts depth
MaxY = max ([maxexp maxth]); % to normalize nanoparticle graph with dimension to the highest peak
MaxDY = max ([maxthDF maxDexp maxthDNPF]);
figure(1) % Nanoparticles Transport in FLuid
bx1 = subplot (2,1,1);
for xi = 1 : length (Depth)-1
    plot (Time, (ConcNP(:,find(NPDepth == Depth (xi+1))-1)),'Color',[0.15*optimj, 0,1])
    hold on
end
```

```
  xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Nanoparticle experimental data'); axis
  tight;

bx2 = subplot (2,1,2);
plot(t/KT,OmegaVfinitial*OmegaNPF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
hold on
xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Nanoparticle model Output'); axis tight
bx1.YLim = [0,MaxY]; bx2.YLim = [0,MaxY];
bx2.XLim = [0,max(Time)];


figure(2) % pesticide Transport in FLuid
dx1 = subplot (3,1,1);
plot (TimeD/KT, (ConcD(:,optimj)),'Color',[0.15*optimj, 0,1])
hold on
  xlabel('Time [min]'); ylabel('C_D_F [mg/cm^3]');title ('Pesticide in nanoparticle experimental
  data'); axis tight;

dx2 = subplot (3,1,2);
plot(t/KT,OmegaDfinitial*OmegaDNPF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
hold on
xlabel('Time [min]'); ylabel('C_D_F [mg/cm^3]');title ('Pesticide in nanoparticle model output');
axis tight

dx3 = subplot (3,1,3);
plot(t/KT,OmegaDfinitial*OmegaDF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
hold on
xlabel('Time [min]'); ylabel('C_F [mg/cm^3]');title (' Free pesticide model output'); axis tight
dx1.YLim = [0,MaxDY]; dx2.YLim = [0,MaxDY]; dx3.YLim = [0,MaxDY];
dx1.XLim = [0,max(Time)];  dx2.XLim = [0,max(Time)]; dx3.XLim = [0,max(Time)];

%% Exporting results in an Excel sheet
EXP = [Time, ConcD]';
Time_m = (t/KT)';
Model(optimj,1:length(Time_m))= Time_m;
ModelD(optimj,1:length(Time_m))= Time_m;
Model(length(Depth)-1+optimj ,1:length(OmegaDNPF_x))= OmegaDfinitial*OmegaDNPF_x(:,optimj);
ModelD(length(Depth)-1+optimj ,1:length(OmegaDF_x))= OmegaDfinitial*OmegaDF_x(:,optimj);
xlswrite('DruginParticleExp.xlsx',EXP');% Experimental pesticide in nanoparticle Data
xlswrite('DruginParticlemodel.xlsx',Model'); % Model pesticide in nanoparticles output
xlswrite('Freepesticidemodel.xlsx',ModelD'); % Model free pesticide output
```

```
end

function Error = objfun (params)
    %% UNKNOWN parameter to fit
    DaDNP = params(1) % Rate constant of pesticide release from nanoparticles [min^-1]

    %% Solving the PDEPE
    % options2=odeset('RelTol',1e-5); % sets the relative tolerance to 10^-5
    sol = pdepe(m,@pdeNPSoil,@pdeNPSoilic,@pdeNPSoilbc,x,t);
    OmegaNPF = sol(:,:,1); % model output of Nanoparticles in fluid in time and space
    OmegaDF = sol(:,:,2); % model output of pesticide in fluid in time and space
    OmegaDNPF = sol(:,:,3); % model output of pesticide attached to virus in fluid in time and space

    % OmegaVF as a function of t at the exit of the column
    OmegaNPF_x(:,optimj)=OmegaNPF(:,length(x));

    % OmegaDF as a function of t at the exit of the column
    OmegaDF_x(:,optimj)=OmegaDF(:,length(x));

    % OmegaDNPF as a function of t at the exit of the column
    OmegaDNPF_x(:,optimj)=OmegaDNPF(:,length(x));

      %% MODIFIED Simpson's Integration to calculate area under the curve
    % Total area under the pesticide in fluid at the exit
    C_x = zeros (length(OmegaDNPF_x),1);
    te = 0;
    for num = 1: length (t)
        C_x (num) = (OmegaDNPF_x (num) + OmegaDF_x (num)) ;
    end

    for i = 2:2:length(t)-2
        te = te + 4*C_x (i) + 2*C_x (i+1);
    end

    te = te + C_x (1) + C_x (length(t)) + 4*C_x (length(t)-1);
    areaE = te * (tmax-tmin)/(length(t)-1)/3;
    AUC = areaE;
    masspesticide = AUC*OmegaDfinitial*L*A*E
    massDrug
    % Total area under the pesticide stuck in soil
```

```
        %% Comparing concentrations and calculating error for specific time points
        Cerror(optimj)=0;
        for xi = 1 : length (t)
            Cerror(optimj) = Cerror(optimj)+ (OmegaDNPF_x(xi,optimj)-FittedConcD(xi,:))^2;
        end
        Error=Cerror(optimj)*10^3

        %% Inner Functions
        function [c,f,s] = pdeNPSoil(x,t,u,DuDx)
            % Defining c, f, and s to solve the PDEPE function
            c = [1;1;1];  % Coefficients in front of DuDt term (none here so all ones)
            f = [Pev_inv(optimj)*DuDx(1);Ped_inv*DuDx(2); Pev_inv(optimj)*DuDx(3) ];  % Coefficients for
            the second derivative
            s = [-DuDx(1) - Dav(optimj)*rsf*u(1);-DuDx(2) - Dad*rsf*u(2) + DaDNP * u(3); -DuDx(3) -
            Dav(optimj)*rsf*u(3) - DaDNP * u(3)]; % Reactive term
        end

        %% Initial Conditions
        function u0 = pdeNPSoilic(x)
            u0 = [0; 0; 0];
        end
        %% Boundary Conditions
        function [pl,ql,pr,qr] = pdeNPSoilbc(xl,ul,xr,ur,t)
            OmegaVfi = 1- 1/(1+exp(-50*(t-tinj)));
            pl = [ul(1)-OmegaVfi; ul(2); ul(3)-OmegaVfi];
            ql = [0;0;0];
            pr = [0;0;0];
            qr = [1;1;1];
        end
    end

%%
WarnWave = [sin(1:.6:400), sin(1:.7:400), sin(1:.4:400)]; % Produces a sound alert when code is done
running
Audio = audioplayer(WarnWave, 22050);
play(Audio);
toc

end
```

Part 4: Theoretical treatment of a crop infected by nematodes

```
    % Last edited: 09/04/18
    %% Scenario:
    % Root-knot nematodes have infected the roots of a crop.
    % Analysis of the soil reveals that their highest density is located 24 cm deep from the surface.
    % To treat the crop, the drug Abamectin is to be used.The IC50 of Abamectin is 1.309 x 10^-4 mg.cm^-3.
    % The TMGMV-Abamectin formulation is explored.


    %QUESTION: what is the concentration of TMGMV-Abamectin that must be applied on the crop to reach the
     IC50 concentration 24 cm deep in the soil?
    %%

function NematodeTreatment
    tic
    clc
    clear

    %% Defining the constants
    E = 0.45; % Volume fraction of fluid  [1 = 100% fluid]
    rsf = (1-E)/E; % Particle-to-fluid volume ratio [dimensionless]
    Phi = 3401; % Adsorption surface per particle volume[cm^-1]
    radius = 1.4; % Radius of the soil column [cm]
    A = pi*radius^2; % Cross sectional Area of the column[cm^2]
    volumeinj = 0.3; % Volume of particle injection [cm^3]
    Lz=24; % Target location [cm]
    MWDrug = 873; % Molecular weight of Abamectin (g/mol)
    ICfifty = 1.309 * 10^-4; % Concentration of Abamectin that  must be
    reached at Lz [mg/cm^3]
    moleDrug = 1e-008 ; % (TMGMV) moles of Abamectin in 1 mg of TMGMV (mol)
    massDrug = MWDrug * moleDrug * 1000;  % Mass of pesticide injected in the
    system [mg]

    %% Optomized parameters of Nanoparticle and pesticide
    NPParamData = xlsread('TMGMVaverage'); % Importing the parameter
    optimizations from nanoparticles
    DrugParamData = xlsread('DRUGaverage'); % Importing the parameter
    optimizations from pesticide
    Time = (0:.1:60); %irrigation for one hour
```

```
Q = [0.1 0.2 0.5 1 2];
for fl = 1: length (Q)
Pev_inv = NPParamData (1); % Peclet number of nanoparticle
Dav = NPParamData (2); % a Damkohler number of nanoparticle
Ped_inv1 = DrugParamData (:,1);% Peclet number of pesticide
Dad1 = DrugParamData(:,2)*(A/0.7854); % a Damkohler number
kd = (Dad1*Q(fl))/(2*A*E*Phi);
Dd = (Q(fl)*2*Ped_inv1)/(A*E);
DaDNP = NPParamData (3); % a a Damkohler number of pesticide release from nanoparticle

%% Parameters Estimation STARTS
%% Defining dimensionless time and length for the PDEPE function
L = 300; % Depth of the nematodes
Dad = (kd*L*A*E*Phi)/Q(fl);
Ped_inv = (Dd*A*E)/(Q(fl)*L);
KT= Q(fl)/(L*A*E); % Dimensionless constant of time
tinj=volumeinj/Q(fl)/2*KT; % Initial time of particle injection into the system
TimeD = Time * KT; % Dimensionless time
m = 0; % The symmetry of the problem. m can be slab = 0, cylindrical = 1, or spherical = 2.
x = linspace(0,1,1000); % Dimensionless
tmin = 0;
tmax = max(TimeD);
t = linspace(tmin,tmax,1001); % Dimensionless


      %% UNKNOWN parameter to fit
      maxi = 1.5; % [mg]
      increment = .2;
      j = 1;
      Cerror = 0;
      AUC = 0;
      for iparam = .2:increment:maxi
        massparticle = iparam; % Initial concentration of nanoparticle applied
        massparticles(j) = massparticle; % [mg]
        OmegaVfinitial = massparticle/volumeinj; % [mg/cm^3]
        OmegaDfinitial = (massDrug/volumeinj) * massparticle ; % [mg/cm^3]

        %% Solving the PDEPE
        % options2=odeset('RelTol',1e-5); % sets the relative tolerance to 10^-5
        sol = pdepe(m,@pdeNPSoil,@pdeNPSoilic,@pdeNPSoilbc,x,t);
        OmegaNPF = sol(:,:,1); % model output of Nanoparticles in fluid in time and space
```

```
OmegaDF = sol(:,:,2); % model output of pesticide in fluid in time and space
OmegaDNPF = sol(:,:,3); % model output of pesticide attached to virus in fluid in time and
space
OmegaNPS = sol (:,:,4); % model output of Nanoparticles bound to soil in time and space
OmegaDS = sol (:,:,5); % model output of pesticide bound to soil in time and space


%calculate length array position for Lz

nLz=round((length(x)-1)*Lz/L);

%calculate variables as a function of t at z=Lz

% OmegaNPF as a function of t at Lz
OmegaNPF_x(:)=OmegaNPF(:,nLz);

% OmegaDF as a function of t at Lz
OmegaDF_x(:)=OmegaDF(:,nLz);

% OmegaDNPF as a function of t at L
OmegaDNPF_x(:)=OmegaDNPF(:,nLz);

 % OmegaNPS as a function of t at Lz
OmegaNPS_x(:)=OmegaNPS(:,nLz);

% OmegaDS as a function of t at Lz
OmegaDS_x(:)=OmegaDS(:,nLz);


 %% MODIFIED Simpson's Integration to calculate area under the curve
% Total area under the pesticide in fluid
C_x = zeros (length(OmegaDNPF_x),1);
Cmin = 0;
te = 0;
for num = 1: length (t)
    C_x (num) = (OmegaDNPF_x (num) + OmegaDF_x(num))*OmegaDfinitial ; %C_x is dimensional
end

for i = 2:2:length(t)-2
    te = te + 4*C_x (i) + 2*C_x (i+1);
end
```

```
    te = te + C_x (1) + C_x (length(t)) + 4*C_x (length(t)-1);
    area = te * (tmax-tmin)/(length(t)-1)/3/KT; %area is dimensional

    tnematode = 24*max(Time);
    AUC(j) = area
    Cerror(j) = ((AUC(j)-ICfifty*tnematode)^2 )*1000
    j = j+1;


  end
  minerror = min(Cerror);
  masstoinject(fl) = massparticles(find (Cerror == minerror))/A
  MinError (fl) = Cerror(find (Cerror == minerror))
   for i = 1: length (massparticles)
    IC1 (i) = ICfifty;
   end
    figure (3)
    plot (massparticles, IC1*tnematode,'r')
    hold on
    plot (massparticles, AUC,'b')
    hold off

       %% Figures
 figure(1) % Nanoparticles Transport in FLuid
bx1 = subplot (5,1,1);
plot(t/KT,OmegaVfinitial*OmegaNPF_x);
xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title (' Free nanoparticle model Output');
axis tight
bx2 = subplot (5,1,2);
plot(t/KT,OmegaVfinitial*OmegaNPS_x);
xlabel('Time [min]'); ylabel('\Omega_N_P_S [mg/cm^3]');title ('bound nanoparticle model Output');
axis tight
bx3 = subplot (5,1,3);
plot(t/KT,OmegaDfinitial*OmegaDNPF_x);
xlabel('Time [min]'); ylabel('C_D_N_P_F [mg/cm^3]');title ('Pesticide in nanoparticle model
output'); axis tight
bx4 = subplot (5,1,4);
plot(t/KT,OmegaDfinitial*OmegaDF_x);
xlabel('Time [min]'); ylabel('C_F [mg/cm^3]');title (' Free pesticide model output'); axis tight
bx5 = subplot (5,1,5);
```

```
    plot(t/KT,OmegaDfinitial*OmegaDS_x);
    xlabel('Time [min]'); ylabel('C_S [mg/cm^3]');title (' Bound
    pesticide model output'); axis tight

    totaldrug =  OmegaDfinitial*OmegaDNPF_x + OmegaDfinitial*OmegaDF_x;
    for i = 1: length (t)
        IC (i) = ICfifty;
    end
    figure (2)
    plot (t/KT,IC, 'r')
    hold on
    plot(t/KT,totaldrug, 'k')
    hold off
    xlabel('Time [min]'); ylabel('Pesticide distribution [mg/cm^3]');title (' Treatment of nematodes');
    axis tight

end
        %% Inner Functions
        function [c,f,s] = pdeNPSoil(x,t,u,DuDx)
            % Defining c, f, and s to solve the PDEPE function
            c = [1;1;1;1;1];  % Coefficients in front of DuDt term (none
            here so all ones)
            f = [Pev_inv*DuDx(1);Ped_inv*DuDx(2); Pev_inv*DuDx(3);
            Pev_inv*DuDx(1)/1000; Ped_inv*DuDx(2)/1000 ];  % Coefficients
            for the second derivative
            s = [-DuDx(1) - Dav*rsf*u(1);-DuDx(2) - Dad*rsf*u(2) + DaDNP * u(3); -DuDx(3) -
            Dav*rsf*u(3) - DaDNP * u(3); Dav*u(1); Dad*u(2) ]; % Reactive term
        end

        %% Initial Conditions
        function u0 = pdeNPSoilic(x)
            u0 = [0; 0; 0; 0; 0];
        end
        %% Boundary Conditions
        function [pl,ql,pr,qr] = pdeNPSoilbc(xl,ul,xr,ur,t)
            OmegaVfi = 1- 1/(1+exp(-50*(t-tinj)));
            pl = [ul(1)-OmegaVfi; ul(2); ul(3)-OmegaVfi; ul(4); ul(5)];
            ql = [0;0;0;0;0];
            pr = [0;0;0;0;0];
            qr = [1;1;1;1;1];
        end
```

```
%%
WarnWave = [sin(1:.6:400), sin(1:.7:400), sin(1:.4:400)]; % Produces a sound alert when code is done
running
Audio = audioplayer(WarnWave, 22050);
play(Audio);
toc


end
```

Part 5: Code used to calculate nanoparticle transport through soil based on average value of Dv and k.

Note: Can also be used for the prediction of nanoparticle transport through soil based on 4 cm depth data.

```
function AveragePdepe
tic
clc
clear
%% Defining the constants
E = 0.45; % Volume fraction of fluid  [1 = 100% fluid]
Q = 1.5; % Flow rate [cm3/min]
Phi = 3401; % Adsorption surface per particle volume[cm^-1]
radius = 1.4; % Radius of the soil column [cm]
A = pi*radius^2; % Cross sectional Area of the column[cm^2]
massparticle = 1; % mass of particle injected in the system [mg]
volumeinj = 0.3; %volume of particle injection [cm^3]
OmegaVfinitial = massparticle/volumeinj; % [mg/cm^3]

%% Estimated Paramters
Kvfp = 3.73*10^-5; % Rate constant of particle absorbation to soil [min^-1]
Dv = 3.02; % Dispersion coefficient of nanoparticles in fluid

%% Experimental Data
ExpData = xlsread('ExperimentalDataMatlab_TMGMVnodelay'); % Importing the experimental data
```

```
%first row is the depth value
Depth = ExpData (1,:,:); % creating the array of soil depth data
Time = ExpData (2:end,1,:); % creating the array of time data.
MassNP = ExpData (2:end,2:end,:); % creating the array of experimental mass of nanoparticles
ConcNP = MassNP /500; % converting the mass into concentration of nanoparticles

for optimj = 1: length(Depth)-1
    %% Non dimensionalization of the time and length
    L = Depth(optimj+1);
    KT= Q/(L*A*E); % dimensionless constant of time
    tinj=volumeinj/Q/2*KT; % Initial time of particle injection into the system%% Non-Dimensionalization of
Experimental Data
    TimeD = Time * KT; % Dimensionless time
    depthD = Depth/L;
    %% Defining dimensionless time and length for the PDEPE function
    m = 0; % the symmetry of the problem. m can be slab = 0, cylindrical = 1, or spherical = 2.
    x = linspace(0,1,1000); % dimensionless
    tmin = 0;
    tmax = ceil(max(TimeD));
    t = linspace(tmin,tmax,1001); % dimensionless
    Index = round(1+ TimeD./(tmax-tmin)*(length (t)-1));

    %% Experimental data as a function of t for specific x values
    ConcNPD = ConcNP./OmegaVfinitial; % Dimensionless concentration
    for i = 1: length (Time)-1
        OmegaNPF_exp(i+1,:)=ConcNPD(i+1,:); %the first index is depth row
    end

    %% Defining the Dimensionless Parameters
    % Equation 1
    KA = (A*E*Dv)/(Q*L);
    KB = (L*A*Kvfp*Phi*(1-E))/(Q);
    %% Solving the PDEPE

    % options2=odeset('RelTol',1e-5); % sets the relative tolerance to 10^-5
    sol = pdepe(m,@pdeNPSoil,@pdeNPSoilic,@pdeNPSoilbc,x,t);
    OmegaNPF = sol(:,:,1); % model output of Nanoparticles in fluid in time and space

    % OmegaVF as a function of t at the exit of the column
    OmegaNPF_x(:,optimj)=OmegaNPF(:,length(x));
```

```
    %% Figures of the results
    maxth = max(OmegaVfinitial.*OmegaNPF_x(:,1)); maxexp = max(smooth(ConcNP(:,1))); % the highest peak is
the firts depth
    MaxY = max ([maxexp maxth]); % to normalize graph with dimension to the highest peak

    maxthD = max(OmegaNPF_x(:,1)); maxexpD = max(smooth(ConcNPD(:,1)));
    MaxYD = max ([maxexpD maxthD]); % to normalize dimensionless graphs to the highest peak

    figure(1) % With Dimensions
    bx1 = subplot (2,1,1);
    plot (TimeD/KT, (ConcNP(:,optimj)),'Color',[0.15*optimj, 0,1])
    hold on
    xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Experimental Data'); axis tight;
    legend (int2str(optimj),'Location','northeast');legend ('boxoff')

    bx2 = subplot (2,1,2);
    plot(t/KT,OmegaVfinitial*OmegaNPF_x(:,optimj),'Color',[0.15*optimj, 0,1]);
    hold on
    xlabel('Time [min]'); ylabel('\Omega_N_P_F [mg/cm^3]');title ('Model Output'); axis tight
    legend(int2str(optimj),'Location','northeast');legend ('boxoff')
    bx1.YLim = [0,MaxY]; bx2.YLim = [0,MaxY];
    bx2.XLim = [0,32];

    EXP = [Time, ConcNP]';
    Time_m = (t/KT)';
    Model(optimj,1:length(Time_m))= Time_m;
    Model(length(Depth)-1+optimj ,1:length(OmegaNPF_x))= OmegaVfinitial*OmegaNPF_x(:,optimj);
    xlswrite('Average.xlsx',Model');
end

%% Inner Functions
    function [c,f,s] = pdeNPSoil(x,t,u,DuDx)
        % Defining c, f, and s to solve the PDEPE function
        c = [1];  % coefficients in front of DuDt term (none here so all ones)
        f = [KA*DuDx(1)];  % coefficients for the second derivative
        s = [-DuDx(1) - KB*(u(1))];
    end
%% Initial Conditions
    function u0 = pdeNPSoilic(x)
        u0 = [0];
    end
```

```
%% Boundary conditions
    function [pl,ql,pr,qr] = pdeNPSoilbc(xl,ul,xr,ur,t)
        OmegaVfi = 1- 1/(1+exp(-50*(t-tinj)));
        pl = [ul(1)-OmegaVfi];
        ql = [0];
        pr = [0];
        qr = [1];

    end
WarnWave = [sin(1:.6:400), sin(1:.7:400), sin(1:.4:400)]; % produces a sound alert when code is done
running
Audio = audioplayer(WarnWave, 22050);
play(Audio);
toc
end
```

Supplementary Table 5. Optimized parameters.

k_{NPS} [cm min^{-1}]: rate constant of nanoparticle absorption to soil.
D_{NP} [cm^2 min^{-1}]: dispersion constant of nanoparticles in the interstitial space.
k_{PS} [cm min^{-1}]: rate constant of pesticide absorption to soil.
D_P [cm^2 min^{-1}]: dispersion constant of pesticide in the interstitial space.
k_{PF} [min^{-1}]: rate constant of pesticide release from nanoparticles.

		TMGMV	**CPMV**	**PhMV**	**MSNP**	**PLGA**
2 cm	k_{NPS}			9.69 x 10^{-05}	7.00 x 10^{-2}	9.81 x 10^{-05}
	D_{NP}			0.72	0.50	0.51
4 cm	k_{NPS}	4.53 x 10^{-05}	1.53 x 10^{-05}	1.07 x 10^{-4}	3.95 x 10^{-2}	1.22 x 10^{-4}
	D_{NP}	1.55	1.35	0.75	1.43	1.60
6 cm	k_{NPS}				2.13 x 10^{-1}	8.92 x 10^{-5}
	D_{NP}				2.07	0.98
8 cm	k_{NPS}	2.81 x 10^{-05}	1.88 x 10^{-05}		1.7 x 10^{-1}	1.06 x 10^{-4}
	D_{NP}	1.34	1.44		2.9758	1.08
12 cm	k_{NPS}	5.36 x 10^{-05}	3.89 x 10^{-06}			
	D_{NP}	5.08	2.71			
16 cm	k_{NPS}	3.92 x 10^{-05}	5.11 x 10^{-05}			
	D_{NP}	4.91	4.42			
20 cm	k_{NPS}	2.07 x 10^{-05}	2.58 x 10^{-05}			
	D_{NP}	2.22	2.95			
30 cm	k_{NPS}	4.26 x 10^{-05}	5.39 x 10^{-05}			
	D_{NP}	6.39	10.17			
Average	k_{NPS}	3.82 x 10^{-05}	2.3 x 10^{-05}	1.02 x 10^{-04}	1.236 x 10^{-01}	1.04 x 10^{-04}
	D_{NP}	3.58	2.57	0.73	1.75	1.04
STD	k_{NPS}	1.19 x 10^{-05}	1.76 x 10^{-05}	6.93 x 10^{-06}	8.12 x 10^{-02}	1.4 x 10^{-05}
	D_{NP}	2.14	1.26	0.02	1.04	0.45

		Cy5
2 cm	k_{PS}	1.37 x 10^{-04}
	D_p	1.73

		TMGMV*Cy5	CPMV*Cy5	PhMV*Cy5	MSNP*Cy5
2 cm	k_{PF}			5.18×10^{-03}	5.08×10^{-05}
4 cm	k_{PF}	5.89×10^{-04}	2.27×10^{-04}	1.26×10^{-03}	3.75×10^{-04}
6 cm	k_{PF}				1.18×10^{-04}
16 cm	k_{PF}		5.46×10^{-04}		
30 cm	k_{PF}		3.49×10^{-04}		
Average	k_{PF}	5.89×10^{-04}	1.05×10^{-03}	3.22×10^{-03}	2.03×10^{-04}
STD	k_{PF}	NA	8.63×10^{-04}	1.96×10^{-03}	1.33×10^{-04}

To quantify how well the computational outputs matched the empirical data, we calculated the difference (error) as follows:

$$\frac{\partial \Omega_{NP}}{\partial t} + \frac{Q}{A\varepsilon}\frac{\partial \Omega_{NP}}{\partial z} = D_{NP}\frac{\partial^2 \Omega_{NP}}{\partial z^2} + \left(\frac{1-\varepsilon}{\varepsilon}\right)\phi R_{NPS} \ , \qquad 0<z<L$$

Error (z) = SUM(OmegaNPF_x(t,z)- OmegaNPF_x(t,z))^2);

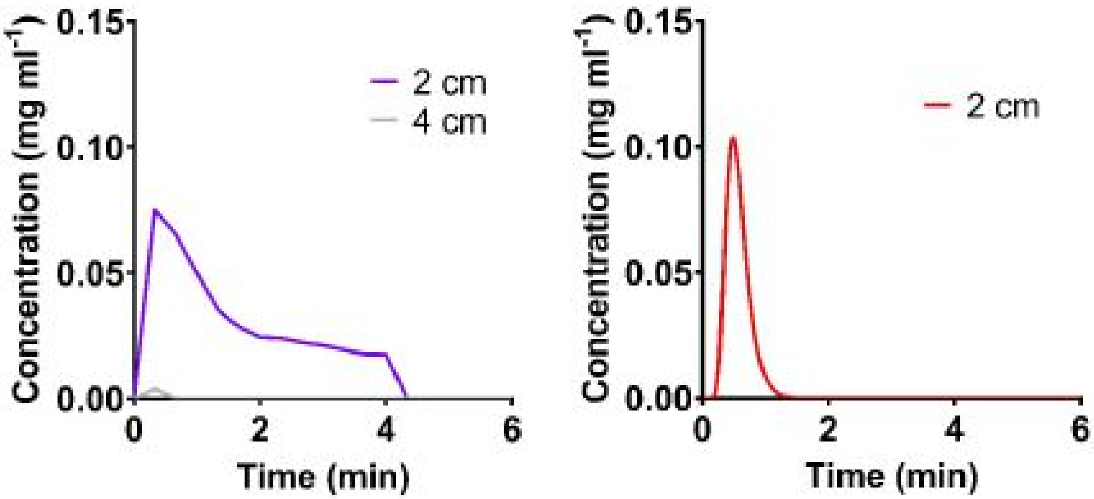
	TMGMV	CPMV	PhMV	MSNP	PLGA
2 cm			3.07×10^{-3}	1.31×10^{-3}	1.67×10^{-3}
4 cm	7.76×10^{-4}	1.03×10^{-3}	7.7×10^{-5}	7.48×10^{-4}	2.98×10^{-4}
6 cm				7.58×10^{-5}	8.56×10^{-5}
8 cm	1.98×10^{-4}	2.06×10^{-3}		1.91×10^{-4}	9.3×10^{-6}
12 cm	9.96×10^{-5}	6.71×10^{-4}			
16 cm	7.48×10^{-4}	3.07×10^{-4}			
20 cm	7.35×10^{-4}	1.86×10^{-3}			
30 cm	2.44×10^{-3}	4.36×10^{-4}			

Error (z) = SUM(OmegaNPF_x(t,z)- OmegaNPF_x(t,z))^2);

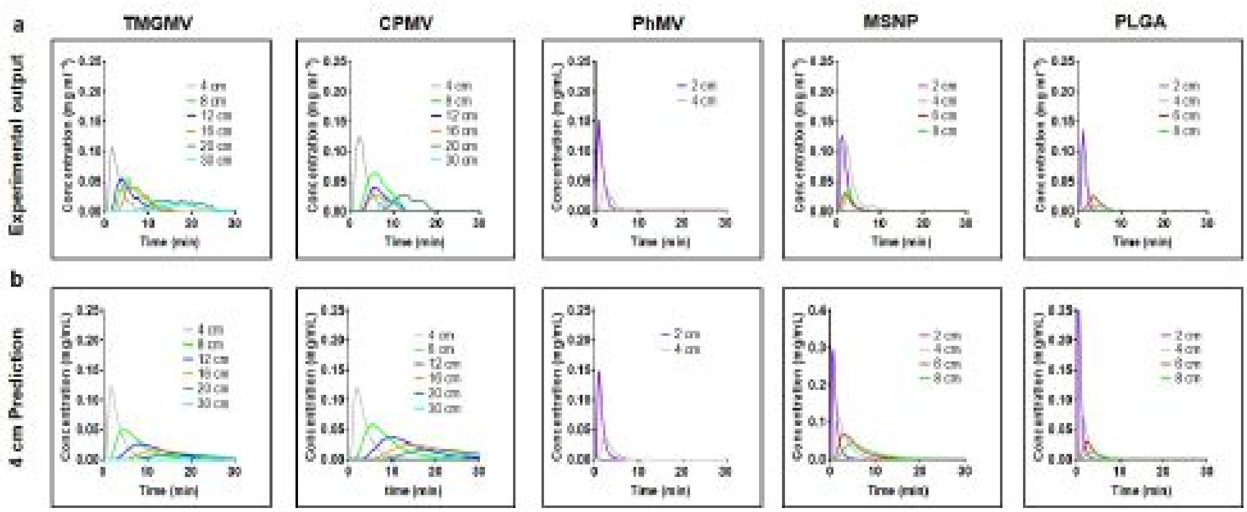
	Cy5
2 cm	7.2×10^{-05}

Error (z) = SUM(OmegaNPF_x(t,z)- OmegaNPF_x(t,z))^2);

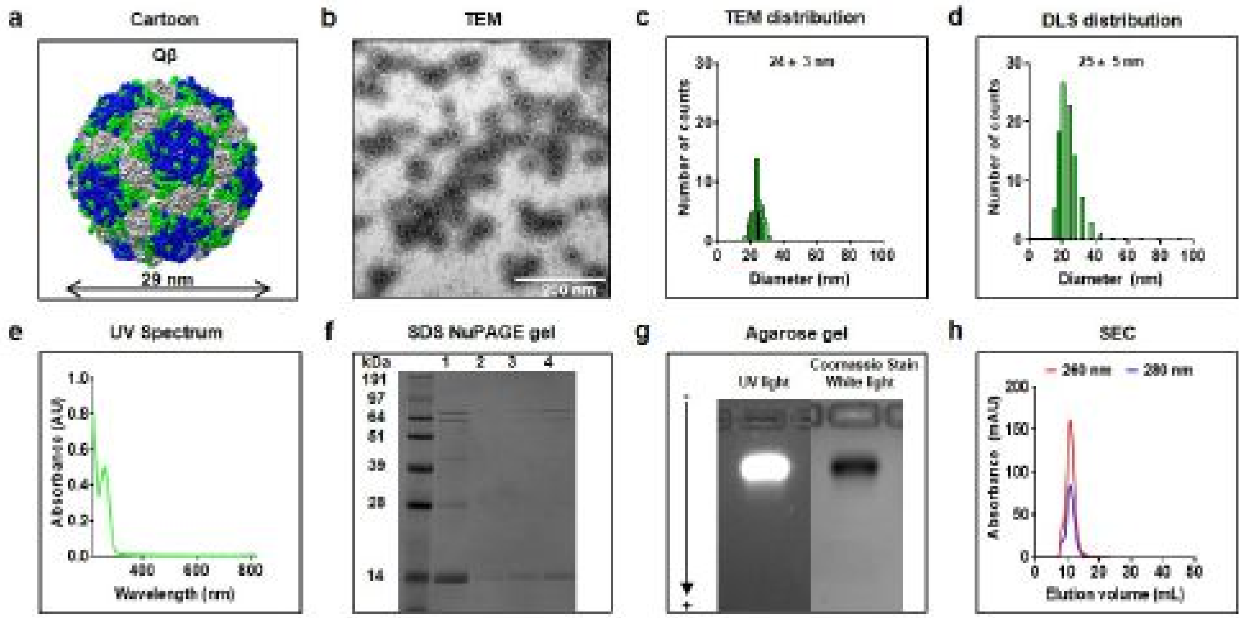
	TMGMV*Cy5	**CPMV*Cy5**	**PhMV*Cy5**	**MSNP*Cy5**
2 cm			6.56 x 10^{-8}	2.27 x 10^{-3}
4 cm	6.59 x 10^{-5}	3.66 x 10^{-8}	1.24 x 10^{-8}	6.66 x 10^{-4}
6 cm				1.65 x 10^{-4}
16 cm		7.82 x 10^{-9}		
30 cm		1.71 x 10^{-8}		



Supplementary Fig. 9 | Free Cy5 from experimental (left) and model output (right).



Supplementary Fig. 10 | Nanoparticle soil transport prediction using a simple 4 cm soil column. a, The empirical output of TMGMV, CPMV, PhMV, MSNP and PLGA is used as a reference. **b,** Computational prediction of nanoparticle transport through soil. $\mathbf{D_{NP}}$ and $\mathbf{k_{NPS}}$ were optimized from the empirical data obtained from the 4 cm soil column.



Supplementary Fig. 11 | Qβ characterization. a, Simplified representation of (1) Qβ (PDB ID: 5KIP). **b,** Corresponding TEM images, Scale bar = 200 nm. **c,** Size distribution analysis of the TEM images. **d,** Particle size distribution obtained by dynamic light scattering (DLS). **e,** UV/Vis spectrum of bare Qβ. **f,** SDS-PAGE gel of Qβ imaged under white light after Coomassie staining. Sample 1, 2, 3, and 4 correspond to 10, 0.5, 1, and 2 μg of Qβ, respectively; higher mobility bands correspond to dimers and multimers. **g,** agarose gel electrophoretic separation of Qβ; gels were imaged under UV light and under white light after Coomassie staining to visualize the RNA (random cellular RNA) and protein. **h**, Size exclusion chromatography of Qβ.

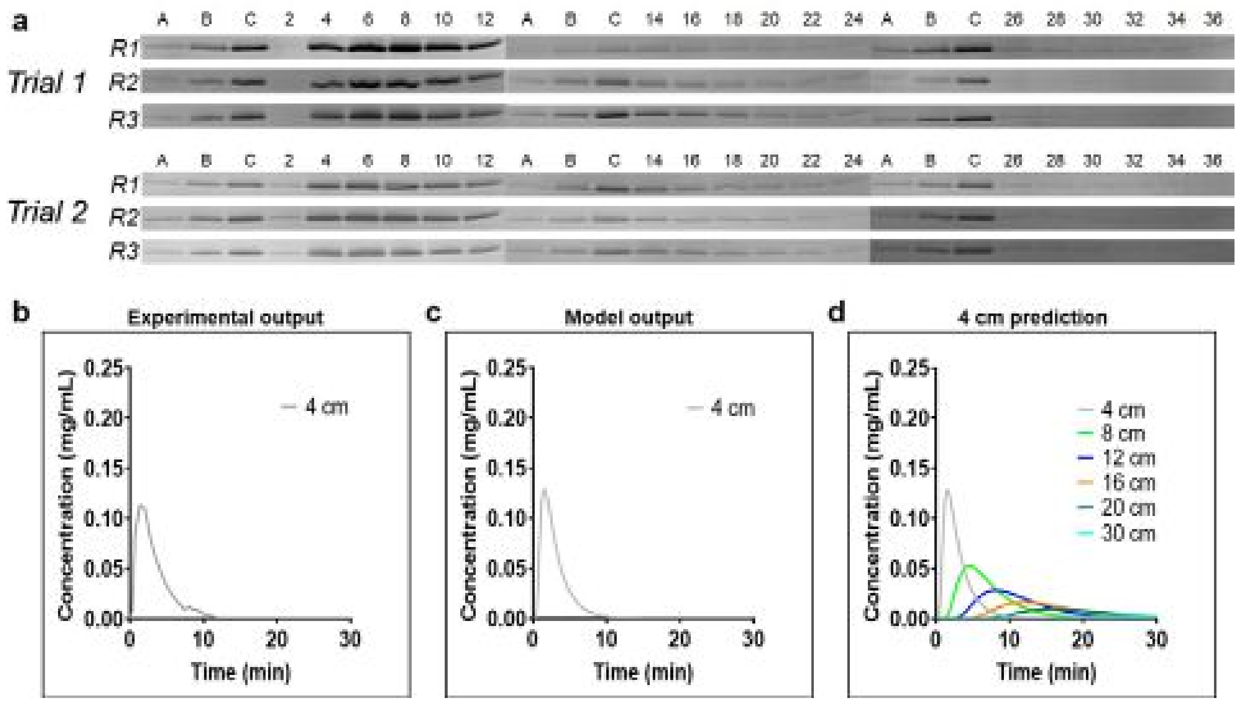
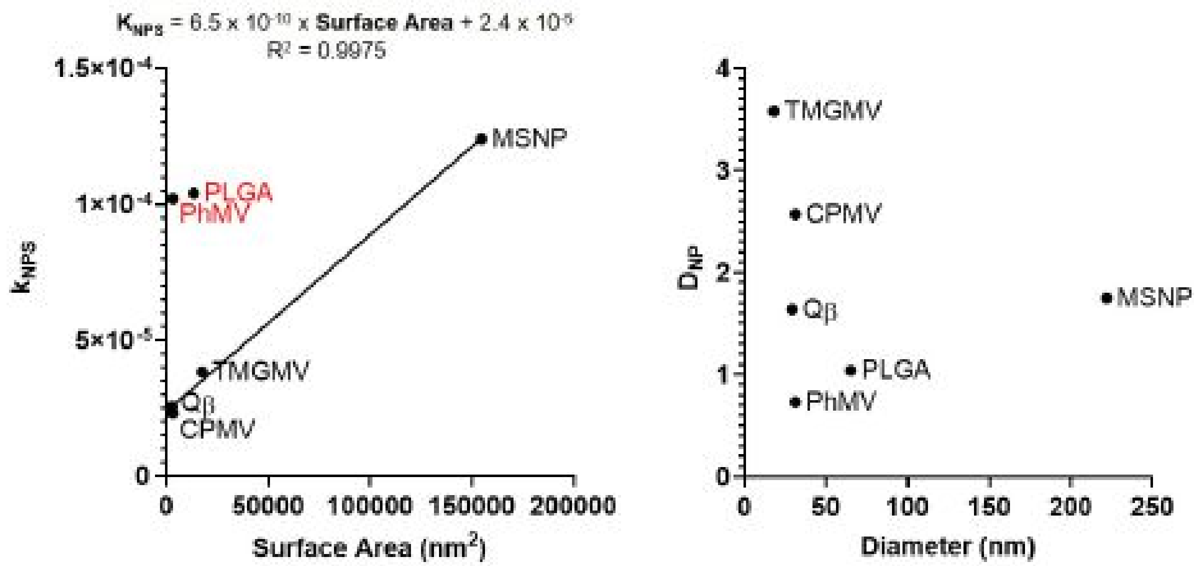


Fig. S12 | Qβ transport through soil. a, SDS-PAGE analysis of Qβ elution fractions exiting the 4 cm-deep soil column. **b,** The empirical output of Qβ is used as a reference. **c,** Computational modelling of Qβ nanoparticle transport through 4 cm of soil. **d**, Computational prediction of nanoparticle transport through soil columns of various length. $\mathbf{D_{NP}}$ and $\mathbf{k_{NPS}}$ were optimized from the empirical data obtained from the 4 cm soil column: $D_{NP\_Q\beta}$ = 1.64 cm^2 min^{-1} and $k_{NP\_Q\beta}$ = 2.54 x 10^{-5} cm min^{-1}.



Supplementary Fig. 13 | Parameter correlation. A linear correlation between the surface area of TMGMV, CPMV, MSNP, and Qβ with the constant of binding rate k_{NPS} was obtained. No linear correlation could be obtained for the dispersion constant D_{NP}.

References:

1. Chariou, P. L. & Steinmetz, N. F. Delivery of pesticides to plant parasitic nematodes using Tobacco mild green mosaic virus as a nanocarrier. *ACS Nano* **11,** 4719–4730 (2017).