

Supporting Information

The measurement of dissolved organic matter fluorescence in aquatic environments: An inter-laboratory comparison

*Kathleen R. Murphy^{*1,2}, Kenna D. Butler³, Robert G.M. Spencer⁴, Colin A. Stedmon⁵, Jennifer R.
Boehme⁶, George R. Aiken^{*3}*

¹The University of New South Wales, School of Civil and Environmental Engineering, Sydney NSW 2052, Australia

²Smithsonian Environmental Research Center, PO Box 28, Edgewater MD 21037, USA

³U.S. Geological Survey, 3215 Marine Street, Boulder, CO 80303, USA

⁴Woods Hole Research Center, 149 Woods Hole Road, Falmouth, MA 02540, USA

⁵National Environmental Research Institute, Aarhus University, Department of Marine Ecology, Frederiksborgvej 399,
DK-4000 Roskilde, Denmark

⁶National Science Foundation, BIO Division of Biological Infrastructure, 4201 Wilson Boulevard, Arlington VA
22230, USA

Prepared for: Environmental Science and Technology

The following 36 pages contain 9 tables, 2 figures and 2 MATLAB scripts in 11
appendices, as described in the table of contents.

TABLE OF CONTENTS

| | |
|--|-----|
| APPENDIX A: PARTICIPATING LABORATORIES..... | S3 |
| APPENDIX B: SOURCES OF DATA FOR FIGURES | S4 |
| APPENDIX C: SOURCES OF DATA IN TABLES AND APPENDICES..... | S5 |
| APPENDIX D: INSTRUMENTATION AND PROCEDURES IMPLEMENTED BY PARTICIPATING LABORATORIES | S6 |
| APPENDIX E: SAMPLE CHARACTERISTICS: ABSORBANCE, DOC AND SUVA | S7 |
| APPENDIX F: INNER FILTER EFFECT (IFE) CORRECTION FACTORS | S8 |
| APPENDIX G: FLUORESCENCE INTENSITIES OF BLIND SAMPLES A-H | S9 |
| APPENDIX H: WITHIN-LABORATORY MEASUREMENT PRECISION..... | S10 |
| APPENDIX I: FLUORESCENCE IN REFRIGERATED CONTROL SAMPLES | S11 |
| APPENDIX J: SCANS OF STARNA 6BF REFERENCE BLOCKS | S13 |
| APPENDIX K: THE FDOMCORR TOOLBOX FOR MATLAB | S15 |

Appendix A: Participating laboratories

Table S1: List of laboratories that participated in the current interlaboratory comparison

Water Research Center, University of New South Wales, Australia
National Environmental Research Institute, Aarhus University, Denmark
Centre for Ecology and Hydrology, National Environment Research Council, UK
Faculty of Health and Life Sciences, University of West England, Bristol, UK
School of Geography, Earth & Environmental Sciences, University of Birmingham, UK
School of Marine Science and Technology, University of Newcastle Upon Tyne, UK
College of Marine Science, University of South Florida, USA
Darling Marine Center, University of Maine, USA
Dept. of Civil and Environmental Engineering, University of California Irvine, USA
Dept. of Civil and Environmental Engineering, Arizona State University, USA
Dept. of Ecology, Evolution and Behavior, University of Minnesota, USA
Dept. of Environmental, Coastal & Ocean Sciences, University of Massachusetts, USA
Dept. of Natural Resources, University of New Hampshire, USA
Dept. of Oceanography, Texas A&M University at Galveston, USA
Forestry Sciences Laboratory - Juneau, United States Forestry Service, USA
Institute of Arctic and Alpine Research, University of Colorado, USA
Marine Invasions Research Laboratory, Smithsonian Environmental Research Center, USA
Southeast Environmental Research Center, Florida International University, USA
United States Geological Survey - Boulder, USA
United States Geological Survey - Sacramento, USA

Appendix B: Sources of data for figures

Table S2: Sources of data for figures in this paper. Included samples are indicated by Y or by Starna Reference set (S1 or S2). Reasons for exclusions were a) no spectral correction files provided, c) unreliable QS dilution series, d) unable to reproduce spectral corrections, e) missing data for samples E, K, C, N, f) very low signal/noise ratio, g) unusual reference cell, i) faulty scan or reference cell.

| Lab | Corr. Factors | Figure: 2 | 3 | 4A | 4b | 4c | 4c | 5 |
|-----|---------------|------------|------|------|------|------|-----|----|
| | | QS spectra | SR 3 | SR 4 | SR 5 | SR 6 | RSE | |
| 11 | Y | Y | S1 | i | S1 | S1 | S1 | Y |
| 12 | Y | Y | S1 | i | S1 | S1 | S1 | Ye |
| 15 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 22 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 23 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 27 | Y | Y | S2 | S2 | S2 | i | S1 | Y |
| 28 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 33 | Y | Y | S1 | i | S1 | S1 | S1 | Y |
| 36 | Y | Y | a | g | g | g | g | Y |
| 38 | a | a | a | a | a | a | a | a |
| 42 | Y | Y | S1 | i | S1 | S1 | S1 | Y |
| 44 | Y | Y | S1 | i | S1 | i | S1 | Y |
| 47 | Y | Y | S1 | i | S1 | i | S1 | Y |
| 50 | Y | Y | S1 | i | S1 | S1 | S1 | Y |
| 51 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 55 | Y | Y | S2 | S2 | S2 | S2 | S2 | Y |
| 62 | Y | Y | S2 | S2 | S2 | S2 | S2 | c |
| 66 | Y | Y | f | S1 | S1 | S1 | S1 | c |
| 88 | a | a | a | a | a | a | a | a |
| 99 | Y | d | g | g | g | g | g | d |

Appendix C: Sources of data in tables and appendices

Table S3: Sources of data included in tables 1 and 2 and appendices C-F in this paper. Abbreviations for topics in appendix D are DOC (dissolved organic carbon), A(λ) (absorbance), SUVA (specific absorbance), IFE (inner filter corrections). Included labs are indicated by Y; Ym indicates missing samples, Yu indicates exclusion from lab corrected datasets, Ye indicates exclusion from all corrected datasets, Yr indicates exclusion from QS normalized datasets. Reasons for exclusions were a) no spectral correction files provided, b) lab-corrected EEMs were not spectrally corrected or were missing, c) unreliable QS dilution series, d) unreliable spectral correction factors, e) missing data for samples E, K, C, N, f) samples were diluted, g) replicates scanned at different instrument settings, h) negative values excluded, i) not included in set of 'top 4 laboratories' for corresponding parameter, p) analysis time exceeded two weeks. For appendices D and E, labs included in set of 'top 4 laboratories' are shown as Y together with samples included: j) samples A-C and their replicates, k) samples D and H, m) samples E and K, n) samples A and L, o) all samples excluding A and L.

| Table: 1 | | Appendix: C | | E | E | E | F | G | H | H |
|----------|--------|----------------|--------|-----------------|------|------|-------------|-----|-----|---|
| Lab | Ratios | Lab procedures | DOC | A (λ) | SUVA | IFE | Intensities | (a) | (b) | |
| 11 | Y | Y | i | Yn,o | Yn,o | Yn,o | Y | Y | Y | |
| 12 | Ym,e | Y | i | i | i | i | Ym,e | Y | p | |
| 15 | Y | Y | Yj,k,m | i | i | i | Y | Y | Y | |
| 22 | Y | Y | i | i | i | i | Y | Y | Y | |
| 23 | Y | Y | i | i | i | i | h | f | f | |
| 27 | Y | Y | Yj,k,m | i | i | i | Y | Y | Y | |
| 28 | Y | Y | Yj,k,m | i | i | i | Y | Y | Y | |
| 33 | Y | Y | Yj | i | i | i | Y | Y | Y | |
| 36 | Y | Y | i | i | i | i | Y | Y | p | |
| 38 | Yu,a | Y | i | i | i | i | Yu,b | Y | Y | |
| 42 | Yu | Y | i | i | i | i | Y | Y | Y | |
| 44 | Yu | Y | i | Yo | Yo | Yo | Y | Y | p | |
| 47 | Y | Y | Ym | i | i | i | Y | Y | Y | |
| 50 | Y | Y | i | Yn | Yn | Yn | Y | Y | Y | |
| 51 | Y | Y | i | i | i | i | h | Y | Y | |
| 55 | Y | Y | i | i | i | i | Y | Y | Y | |
| 62 | Yu | Y | i | Yn,o | Yn,o | Yn,o | Yr,c | g | g | |
| 66 | Y | Y | i | i | i | i | Yr,c | Y | Y | |
| 88 | Yu,a | Y | Yk | i | i | i | Yu,a | Y | p | |
| 99 | Yu,d | Y | i | Yn | Yo | Yo | Yu,d | Y | p | |

Appendix D: Instrumentation and procedures implemented by participating laboratories

Table S4: Instrumentation and data correction procedures implemented by participating laboratories in this study, showing nine unique combinations of correction steps. Instrument manufacturers (Man.) were PE: Perkin Elmer, VAR: Varian, HJY: Horiba Jobin Yvon, PTI: Photon Technologies International, HIT: Hitachi. Temperature ranges (Temp.) during analysis are shown and correction procedures applied are indicated by black dots: Blank subtraction (BKS), inner filter correction (IFC), excitation correction (ExCor), emission correction (EmCor). Data were normalized (Norm.) to units of QSE (Quinine sulfate) or Raman units (RU) or left as counts or in raw machine units.

| Lab | Model | Man. | Temp. | BKS | IFC | ExCor | EmCor | Norm. |
|-----|--------------|------|-----------|-----|-----|-------|-------|-------|
| 11 | FluoroMax-3 | HJY | 22-23 | • | • | • | • | QSE |
| 12 | Cary Eclipse | VAR | 21 | • | | • | • | |
| 15 | FluoroMax-3 | HJY | 21 | • | • | • | • | RU |
| 22 | LS 50 | PE | 21 | • | • | • | • | |
| 23 | FluoroMax-3 | HJY | 16-17 | • | • | • | • | RU |
| 27 | FluoroMax-4 | HJY | 20 | • | | | | RU |
| 28 | FluoroMax-3 | HJY | 24 | • | • | • | • | RU |
| 33 | Fluorolog 3 | HJY | 20 | • | | • | • | QSE |
| 36 | Cary Eclipse | VAR | 22.2-23.4 | • | • | • | • | RU |
| 38 | Cary Eclipse | VAR | 20 | | | | | |
| 42 | FluoroMax-3 | HJY | 22-24 | • | • | • | • | |
| 44 | FluoroMax-2 | HJY | 20 | • | | • | • | QSE |
| 47 | FluoroMax-3 | HJY | 23.5 | • | • | • | • | RU |
| 50 | FluoroMax-2 | HJY | 20 | • | | • | • | QSE |
| 51 | Cary Eclipse | VAR | 20-22 | | • | • | • | |
| 55 | FluoroMax-4 | HJY | 20 | • | • | • | • | QSE |
| 62 | Cary Eclipse | VAR | 19.5-21 | • | • | • | • | |
| 66 | F-4500 | HIT | 20 | | | • | • | |
| 88 | Cary Eclipse | VAR | 20 | | | | | |
| 99 | QM 4SE | PTI | 24-27.5 | • | • | • | • | RU |

Appendix E: Sample characteristics: absorbance, DOC and SUVA

Table S5: Absorbance (A) and DOC characteristics of samples included in the intercalibration study. DOC for Florida Straight is from the Consensus Reference Materials Project. Other values are means from 4 'most reliable' laboratories as in Appendix C. Measurement uncertainties are $1\times SD$ for A_{254} and DOC, and propagated relative uncertainty for $SUVA_{254}$.

| Sample Name | A_{254} | Absorption coefficient (m^{-1}) | | | | | | | | | DOC (mgL^{-1}) | $SUVA_{254}$ ($\text{LmgC}^{-1}\text{m}^{-1}$) | |
|------------------------------|-----------|-------------------------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------------------------|---|---------------|
| | | 250 nm | 254 nm | 280 nm | 300 nm | 312 nm | 350 nm | 412 nm | 440 nm | 650 nm | | | |
| Pacific Ocean Fulvic Acid | A & L | $0.06\pm6\times10^{-4}$ | 15.54 | 13.98 | 8.04 | 4.94 | 3.81 | 1.87 | 0.62 | 0.41 | 0.02 | 9.24 ± 0.31 | $0.7\pm3.5\%$ |
| Suwannee River Fulvic Acid | B & M | $0.09\pm9\times10^{-4}$ | 22.53 | 21.68 | 16.09 | 12.29 | 10.34 | 5.8 | 1.72 | 1.02 | 0.03 | 2.55 ± 0.13 | $3.7\pm5.2\%$ |
| Pony Lake Fulvic Acid | C & N | $0.11\pm6\times10^{-4}$ | 27.61 | 26.24 | 19.74 | 14.86 | 12.32 | 6.77 | 2.35 | 1.44 | 0.04 | 4.65 ± 0.16 | $2.4\pm3.5\%$ |
| Penobscot River | D & H | $0.16\pm2\times10^{-3}$ | 37.36 | 36.1 | 27.28 | 20.37 | 17.09 | 9.51 | 3.26 | 2.08 | 0.12 | 4.23 ± 0.15 | $3.7\pm3.7\%$ |
| Boulder Creek | E & K | $0.13\pm1\times10^{-3}$ | 33.1 | 30.63 | 23.01 | 17.53 | 14.61 | 7.4 | 2.77 | 1.87 | 0.18 | 8.51 ± 0.28 | $1.6\pm3.5\%$ |
| Hansell CRM - Florida Strait | F & I | $0.00\pm3\times10^{-4}$ | 1.08 | 0.92 | 0.50 | 0.31 | 0.25 | 0.08 | 0.00 | 0.00 | 0.00 | $0.49-0.53^1$ | $0.8\pm7.5\%$ |
| Quinine Sulfate | G & J | $0.01\pm5\times10^{-4}$ | 3.27 | 2.93 | 1.78 | 1.99 | 1.9 | 0.67 | 0.06 | 0.04 | 0.03 | N/A | N/A |

¹ Represents the acceptable range for Hansell CRM according to the Consensus Reference Materials Project (<http://www.rsmas.miami.edu/groups/biogeochem/CRM.html>)

Appendix F: Inner filter effect (IFE) correction factors

Table S6: Correction factors for inner filter effects derived from 'best average' absorbance scans in this study

| Peak | $\lambda_{\text{ex}}/\lambda_{\text{em}}$ (nm) | E & K | A & L | F & I | D & H | C & N | G & J | B & M |
|------|--|-------|-------|-------|-------|-------|-------|-------|
| A | 250/450 | 1.03 | 1.00 | 1.01 | 1.04 | 1.06 | 1.00 | 1.05 |
| C | 350/450 | 1.06 | 1.00 | 1.02 | 1.07 | 1.10 | 1.01 | 1.08 |
| M | 320/410 | 1.12 | 1.01 | 1.08 | 1.15 | 1.22 | 1.02 | 1.19 |
| T | 280/340 | 1.12 | 1.00 | 1.05 | 1.15 | 1.21 | 1.01 | 1.17 |
| B | 275/304 | 1.16 | 1.00 | 1.07 | 1.19 | 1.27 | 1.02 | 1.23 |

Appendix G: Fluorescence intensities of blind samples A-H

Table S7: Fluorescence intensities (mean \pm CV) of EEMs corrected according to the standard algorithms in Raman (RU) and quinine sulfate (QSE) units. Sample size was 13-15 laboratories (see Appendix).

| Sample | SUVA ₂₅₄ | Fluorescence Intensity (RU) | | | | | Fluorescence Intensity (QSE) | | | | | |
|------------------|---------------------|--|-----------------|-----------------|-----------------|-----------------|------------------------------|-------------------|------------------|-----------------|------------------|------------------|
| | | (Lmg C ⁻¹ m ⁻¹) | 250/450 | 350/450 | 320/410 | 280/340 | 275/304 | 250/450 | 350/450 | 320/410 | 280/340 | 275/304 |
| Pacific Ocean FA | A&L | 0.7 | 0.3 \pm 0.17 | 0.15 \pm 0.15 | 0.19 \pm 0.13 | 0.14 \pm 0.17 | 0.57 \pm 0.28 | 24.06 \pm 0.12 | 11.71 \pm 0.07 | 15.2 \pm 0.09 | 11.56 \pm 0.19 | 46.48 \pm 0.34 |
| Suwannee Rr FA | B&M | 3.7 | 0.72 \pm 0.18 | 0.31 \pm 0.16 | 0.27 \pm 0.14 | 0.05 \pm 0.25 | 0.39 \pm 0.14 | 57.61 \pm 0.14 | 24.89 \pm 0.07 | 21.8 \pm 0.1 | 4.18 \pm 0.19 | 31.55 \pm 0.29 |
| Pony Lake FA | C&N | 2.4 | 0.99 \pm 0.26 | 0.18 \pm 0.29 | 0.05 \pm 0.28 | 0.01 \pm 0.79 | 0.35 \pm 0.16 | 78.18 \pm 0.19 | 14.24 \pm 0.17 | 4.05 \pm 0.16 | 0.65 \pm 0.61 | 28.71 \pm 0.31 |
| Penobscot Rr | D&H | 3.7 | 0.74 \pm 0.17 | 0.35 \pm 0.15 | 0.32 \pm 0.12 | 0.08 \pm 0.13 | 0.37 \pm 0.15 | 60.13 \pm 0.13 | 28.16 \pm 0.08 | 25.8 \pm 0.11 | 6.85 \pm 0.17 | 30.67 \pm 0.28 |
| Boulder Creek | E&K | 1.6 | 1.57 \pm 0.17 | 0.99 \pm 0.18 | 0.97 \pm 0.14 | 0.91 \pm 0.12 | 0.86 \pm 0.25 | 126.78 \pm 0.12 | 79.65 \pm 0.05 | 78.1 \pm 0.07 | 74.35 \pm 0.15 | 70.76 \pm 0.32 |
| Florida Strait | F&I | 0.8 | 0.03 \pm 0.63 | 0.01 \pm 0.67 | 0.02 \pm 0.57 | 0.02 \pm 1.53 | 0.34 \pm 0.42 | 2.28 \pm 0.51 | 0.86 \pm 0.54 | 1.18 \pm 0.45 | 1.72 \pm 1.38 | 27.64 \pm 0.5 |
| Quinine sulfate | G&J | - | 0.99 \pm 0.26 | 0.18 \pm 0.29 | 0.05 \pm 0.28 | 0.01 \pm 0.79 | 0.35 \pm 0.16 | 78.18 \pm 0.19 | 14.24 \pm 0.17 | 4.05 \pm 0.16 | 0.65 \pm 0.61 | 28.71 \pm 0.31 |

Appendix H: Within-laboratory measurement precision

Table S8: Comparison of within-laboratory (inter-replicate) measurement precision of study-corrected EEMs for samples A&L , B&M, C&N and E&K at 5 wavelength pairs corresponding with peaks A, C, M, B and T. Table shows the percentage of samples (n = number of laboratories included) for which the coefficient of variation (CV =SD/mean) was below 3, 5 and 10%, in (a) samples analyzed on any date, and (b) samples analyzed within 2 weeks.

| Peak | $\lambda_{\text{ex}}/\lambda_{\text{em}}$ (nm) | CV<3% | CV<5% | CV<10% | CV<3% | CV<5% | CV<10% |
|------|--|------------|-------|--------|------------|-------|--------|
| | | (a) n = 18 | | | (b) n = 13 | | |
| A | 250/450 | 73 | 89 | 99 | 73 | 92 | 98 |
| C | 350/450 | 77 | 94 | 99 | 76 | 94 | 98 |
| M | 320/410 | 76 | 91 | 99 | 78 | 92 | 98 |
| T | 280/340 | 67 | 83 | 96 | 71 | 86 | 96 |
| B | 275/304 | 76 | 93 | 96 | 82 | 94 | 94 |

Appendix I: Fluorescence in refrigerated control samples

Table S9: Time series of fluorescence measurements at peaks A, C, M, T and B and fluorescence index (FI_c : ratio of emissions $\lambda_{em}=470\text{ nm}$ / $\lambda_{em}=520\text{ nm}$ at excitation $\lambda_{ex}=370\text{ nm}$) in refrigerated control samples. Data were measured on a single instrument and represent corrected intensities (in Raman units) from measurements made at 21°C. Sample age is indicated by days analysed elapsed since sample collection or preparation.

| Sample | Age (days) | Analysis date | A | C | M | T | B | FI_c |
|---|------------|---------------|---------|---------|---------|---------|---------|--------|
| | | | 250/450 | 350/450 | 320/410 | 280/340 | 275/304 | |
| Pacific Ocean Fulvic Acid (prepared 14/Apr/08) | | | | | | | | |
| A | 3 | 17-Apr | 0.302 | 0.147 | 0.193 | 0.146 | 0.335 | 1.41 |
| A | 16 | 30-Apr | 0.299 | 0.147 | 0.190 | 0.145 | 0.328 | 1.43 |
| A | 31 | 15-May | 0.297 | 0.145 | 0.192 | 0.142 | 0.334 | 1.42 |
| A | 45 | 29-May | 0.305 | 0.147 | 0.192 | 0.145 | 0.334 | 1.43 |
| L | 4 | 18-Apr | 0.296 | 0.145 | 0.190 | 0.143 | 0.326 | 1.42 |
| L | 17 | 1-May | 0.305 | 0.143 | 0.189 | 0.144 | 0.327 | 1.42 |
| L | 32 | 16-May | 0.291 | 0.139 | 0.184 | 0.138 | 0.324 | 1.44 |
| L | 45 | 29-May | 0.298 | 0.144 | 0.191 | 0.146 | 0.330 | 1.41 |
| Suwannee River Fulvic Acid (prepared 14/Apr/08) | | | | | | | | |
| B | 3 | 17-Apr | 0.636 | 0.299 | 0.263 | 0.047 | 0.000 | 1.25 |
| B | 16 | 30-Apr | 0.643 | 0.296 | 0.259 | 0.045 | 0.000 | 1.24 |
| B | 31 | 15-May | 0.647 | 0.296 | 0.266 | 0.046 | 0.002 | 1.25 |
| B | 45 | 29-May | 0.660 | 0.305 | 0.269 | 0.048 | -0.001 | 1.27 |
| M | 7 | 21-Apr | 0.654 | 0.301 | 0.264 | 0.047 | 0.002 | 1.25 |
| M | 17 | 1-May | 0.640 | 0.297 | 0.260 | 0.044 | -0.004 | 1.25 |
| M | 32 | 16-May | 0.638 | 0.297 | 0.262 | 0.047 | 0.000 | 1.25 |
| M | 45 | 29-May | 0.667 | 0.306 | 0.271 | 0.060 | 0.011 | 1.26 |
| Pony Lake Fulvic Acid (prepared 14/Apr/08 from isolate of lot 1/2006) | | | | | | | | |
| C | 3 | 17-Apr | 0.877 | 0.436 | 0.462 | 0.165 | 0.047 | 1.42 |
| C | 16 | 30-Apr | 0.838 | 0.421 | 0.447 | 0.159 | 0.046 | 1.42 |
| C | 31 | 15-May | 0.856 | 0.431 | 0.459 | 0.167 | 0.051 | 1.42 |
| C | 45 | 29-May | 0.876 | 0.438 | 0.466 | 0.173 | 0.057 | 1.42 |
| N | 7 | 21-Apr | 0.836 | 0.430 | 0.456 | 0.159 | 0.049 | 1.41 |
| N | 17 | 1-May | 0.866 | 0.429 | 0.455 | 0.164 | 0.048 | 1.42 |
| N | 32 | 16-May | 0.859 | 0.427 | 0.456 | 0.165 | 0.050 | 1.42 |
| N | 45 | 29-May | 0.896 | 0.444 | 0.475 | 0.175 | 0.056 | 1.43 |
| Penobscot River water (collected 4/Mar/08) | | | | | | | | |
| D | 44 | 17-Apr | 0.745 | 0.352 | 0.328 | 0.088 | 0.021 | 1.30 |
| D | 57 | 30-Apr | 0.736 | 0.346 | 0.322 | 0.088 | 0.022 | 1.31 |
| D | 72 | 15-May | 0.725 | 0.344 | 0.327 | 0.085 | 0.020 | 1.31 |
| D | 87 | 30-May | 0.773 | 0.362 | 0.342 | 0.084 | 0.020 | 1.31 |
| H | 45 | 18-Apr | 0.641 | 0.337 | 0.304 | 0.077 | 0.017 | 1.30 |
| H | 57 | 30-Apr | 0.617 | 0.329 | 0.296 | 0.074 | 0.016 | 1.30 |
| H | 72 | 15-May | 0.616 | 0.329 | 0.301 | 0.071 | 0.014 | 1.32 |
| H | 87 | 30-May | 0.657 | 0.341 | 0.314 | 0.070 | 0.013 | 1.30 |

| Sample | Age (days) | Analysis date | A | C | M | T | B | FI _c |
|--|------------|---------------|---------|---------|---------|---------|---------|-----------------|
| | | | 250/450 | 350/450 | 320/410 | 280/340 | 275/304 | |
| Boulder Creek water (collected 8/Apr/08) | | | | | | | | |
| E | 9 | 17-Apr | 1.550 | 0.978 | 0.985 | 0.991 | 0.641 | 1.83 |
| E | 22 | 30-Apr | 1.536 | 0.958 | 0.966 | 0.962 | 0.598 | 1.84 |
| E | 37 | 15-May | 1.567 | 0.961 | 0.984 | 0.979 | 0.651 | 1.85 |
| E | 52 | 30-May | 1.660 | 1.015 | 1.042 | 1.023 | 0.682 | 1.84 |
| K | 10 | 18-Apr | 1.341 | 0.923 | 0.907 | 0.852 | 0.512 | 1.83 |
| K | 22 | 30-Apr | 1.332 | 0.927 | 0.909 | 0.871 | 0.501 | 1.83 |
| K | 38 | 16-May | 1.365 | 0.931 | 0.924 | 0.857 | 0.537 | 1.84 |
| K | 52 | 30-May | 1.409 | 0.956 | 0.962 | 0.884 | 0.560 | 1.85 |
| Marine Standard (sealed ampules from lot 12/2007) | | | | | | | | |
| F | 128 | 21-Apr | 0.019 | 0.009 | 0.012 | 0.012 | 0.013 | 1.47 |
| F | 138 | 1-May | 0.018 | 0.008 | 0.012 | 0.006 | 0.007 | 1.52 |
| F | 152 | 15-May | 0.016 | 0.008 | 0.011 | 0.006 | 0.006 | 1.51 |
| F | 167 | 30-May | 0.018 | 0.009 | 0.013 | 0.007 | 0.010 | 1.47 |
| I | 128 | 21-Apr | 0.019 | 0.009 | 0.012 | 0.008 | 0.014 | 1.51 |
| I | 138 | 1-May | 0.036 | 0.011 | 0.017 | 0.010 | 0.008 | 1.68 |
| I | 152 | 15-May | 0.016 | 0.008 | 0.012 | 0.006 | 0.005 | 1.56 |
| I | 167 | 30-May | 0.017 | 0.008 | 0.012 | 0.006 | 0.003 | 1.60 |
| Quinine Sulfate (prepared 14/Apr/08) | | | | | | | | |
| G | 3 | 17-Apr | 0.996 | 0.148 | 0.045 | 0.004 | 0.002 | - |
| G | 16 | 30-Apr | 1.010 | 0.168 | 0.054 | 0.003 | 0.001 | - |
| G | 31 | 15-May | 0.936 | 0.156 | 0.055 | 0.003 | 0.003 | - |
| G | 45 | 29-May | 1.081 | 0.218 | 0.071 | 0.005 | 0.000 | - |
| J | 4 | 18-Apr | 1.003 | 0.159 | 0.048 | 0.003 | 0.001 | - |
| J | 17 | 1-May | 1.042 | 0.176 | 0.053 | 0.003 | 0.001 | - |
| J | 32 | 16-May | 0.913 | 0.142 | 0.047 | 0.003 | 0.000 | - |
| J | 45 | 29-May | 0.980 | 0.169 | 0.057 | 0.003 | 0.001 | - |

Appendix J: Scans of Starna 6BF reference blocks

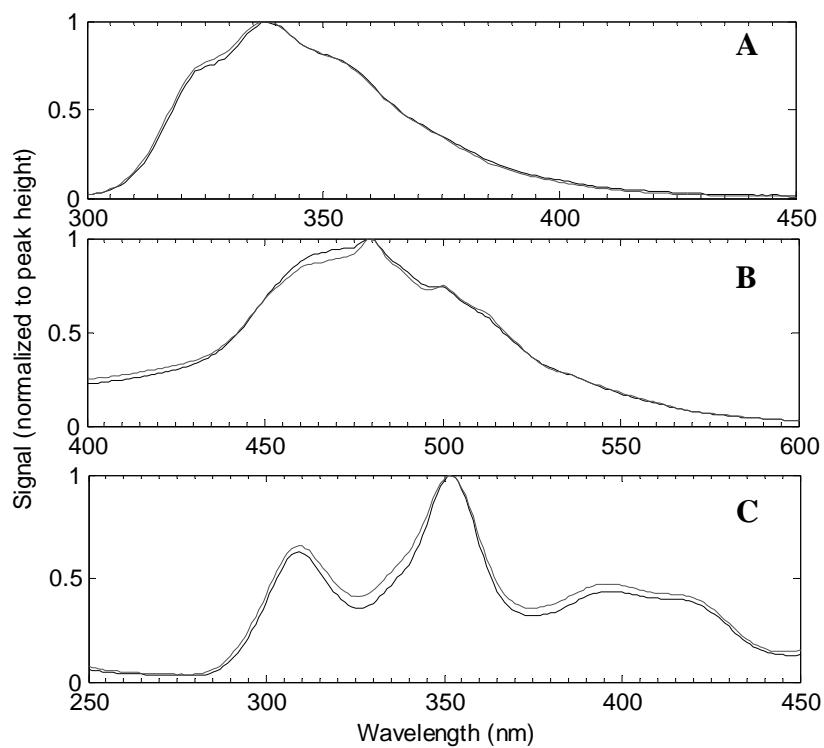


Figure S1: Corrected scans of the two circulated sets (black and grey lines) of Starna polymer fluorescence reference blocks, scanned on a single instrument on the same day at the conclusion of the study. A) Emission scan of SR3: p-Terphenyl ($\lambda_{\text{ex}}=265 \text{ nm}$); B) Emission scan of SR5: Compound 610 ($\lambda_{\text{ex}}=350 \text{ nm}$); and D) Excitation scan of SR6: Rhodamine B ($\lambda_{\text{em}}=562 \text{ nm}$).

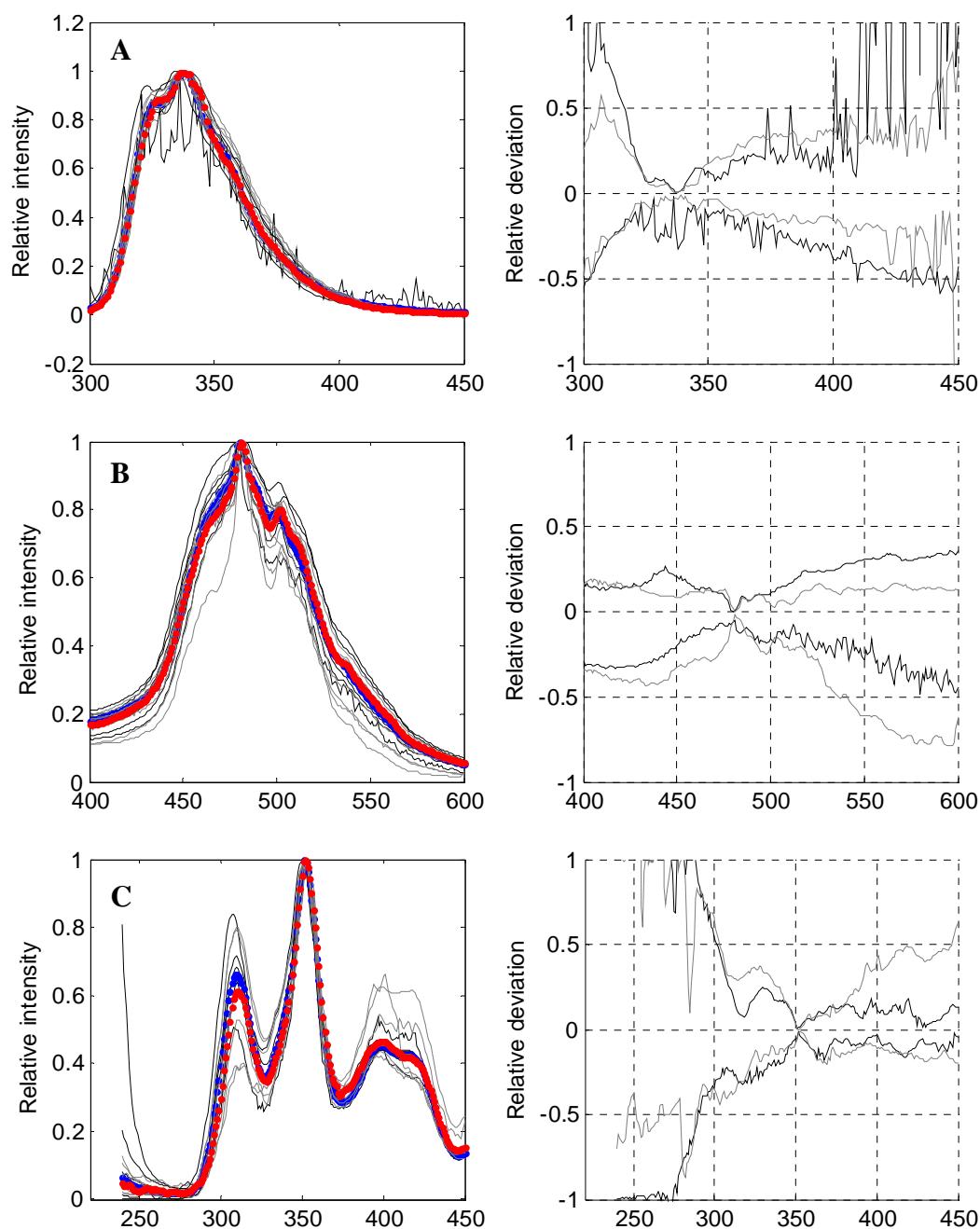


Figure S2: Comparison of study-corrected scans of Starna reference blocks for A) Emission scan of SR3: p-Terphenyl ($\lambda_{ex}=265$ nm, n=16); B) Emission scan of SR5: Compound 610 ($\lambda_{ex}=350$ nm, n=16); and C) Excitation scan of SR6: Rhodamine B ($\lambda_{em}=562$ nm, n=13). Left hand plots show scans normalized to maximum peak height as in Figure 4 of the paper, and right hand plots show maximum positive and negative relative deviation from median scans. Black and grey lines indicate the two different sets of reference cells, with blue and red dots respectively indicating median scans.

Appendix K: The FDOMcorr toolbox for MATLAB

The code for two MATLAB executable files (M-files) is provided here. Check for FAQs, revisions, and related open-source software on the internet: <http://www.models.life.ku.dk/algorithms>, following the links to the FDOMcorr toolbox.

1. ReadInEEMs.m : loads individual EEMs into a 3D matrix in preparation for further processing.
2. FDOMcorrect.m : implements the EEM correction algorithms presented in this paper.

K1: Importing EEMs into MATLAB

The MATLAB function "ReadInEEMs.m" loads individual EEMs into a 3D matrix in preparation for further processing. The program assumes that all files of the specified type (xls, xlsx or csv) that are located in the current directory are EEMs and will attempt to (a) load them one by one in alphabetical order, then (b) merge them into a single 3D matrix. Files that are not EEMs or that are not in the appropriate format will cause the program to abort by error.

Copy and paste the following script into a single M-file named ReadInEEMs and place it in the MATLAB working directory or in another directory in the current path. Confirm that the file path can be located from the current directory using:

```
>which ReadInEEMs
```

Instructions for using the M-file are obtained by typing:

```
>help ReadInEEMs
```

Examples

Read in a dataset of 55 uncorrected EEMs from the directory "c:/data/eem files/Oct1". The EEMs were exported from a Varian Cary Eclipse fluorometer and exist as *.csv files with two rows of text and emission wavelengths in every second column. Plots of each EEM are shown and raw data are automatically saved to the current directory:

```
>> cd 'c:/data/eem files/Oct1'  
>> [X,Emmat,Exmat,filelist,outdata]=ReadInEEMs(2,'csv','A3..CD113',[0 1],1,1);
```

Read in a dataset of 18 uncorrected EEMs from the current directory. The EEMs were exported from a FluoroMax 3 to *.xls files with excitation wavelengths in the first row and emission wavelengths in the first column. No plots or automatically saved data files:

```
>> [X,Emmat,Exmat,filelist,outdata]= ReadInEEMs(1,'xls','A1..AT80',[1 1],0,0);
```

Read in a dataset of uncorrected EEMs from the current directory. The EEMs are *.csv files with no excitation or emission wavelengths. Don't show plots but do export the data:

```
>> [X,Emmat,Exmat,filelist,outdata]= ReadInEEMs(1,'xls','A1..AT80',[1 1],0,1);
```

If you have selected the option to export data, but numerical Ex or Em headers were missing or removed, the program will prompt you to enter the wavelength range for Ex and Em. For example:

```
Specify the excitation wavelength range, e.g. 250:5:450: >> 250:5:400  
Specify the emission wavelength range, e.g. 300:2:600: >> 300:2:550
```

```

function [X,Emmat,Exmat,filelist,outdata]=ReadInEEMs(type,format,range,headers,display,outdat)
%function [X,Emmat,Exmat,filelist]=ReadInEEMs(type,format,headers,display,outdat);
%Reads in all files of specified format (xls,xlsx or csv) in the current directory
%and converts them to a 3D matrix of EEMs. Remove non-EEM files of specified format
%from the current directory before running.
%The files will be loaded in alphabetical order (see outputted file list).
%Input EEMs assumed to each be of the same size, type and format with same Ex and Em
%
%%%%%
%Input variables
%%%%%
%1. type = 1 or 2, defined as below
%    1: Ex in columns and Em in rows (usual output from Fluoromax, Hitachi etc.)
%    2: Ex in columns and Em in rows, with Em headers displayed in every second column (Varian)
%
%2. format = 'csv' , 'xls', 'xlsx';
%           by default, data will be imported from worksheet of name 'Sheet1' in excel
%
%3. range = Range of cells in the raw EEMs that contain numbers (NOT TEXT).
% For most EEM files, there are no text rows that need be excluded e.g. range ='A1..AA500'
% For varian files, usually the first two rows must be excluded as well as any rows of text below the EEMs, e.g. range ='A3..CD113'
% The chosen range affects the designation of headers, see below.
%
%4. headers: presence/absence for Ex and Em,reflecting data excluded from the range
%    [1 1]: present for Ex and Em - this is usual if numerical Ex and Em headers are present and were not excluded from the range
%    [0 1]: present for Em but not Ex - this is usual for Varian files once text headers are excluded from range
%    [1 0]: present for Ex but not Em
%    [0 0]: absent for Ex and Em
%
%5. display = 1 to show plots of EEMs, 0 for no plots
%
%6. outdat = 1 to save data to file in the current directory, or 0 otherwise. Saved data are:
%    (a) outdata: intensities at common wavelength pairs (A, C, M,T,B) saved to an excel workbook (OutData_FDOM.xls)
%    (b) X,Emmat,Exmat,filelist and outdata saved to a matlab file (RawData_FDOM.mat)
%
%%%%%
%Output variables
%%%%%
% X=3D matrix of EEMs
% Emmat=matrix of Em corresponding with individual EEMs
% Exmat=matrix of Ex corresponding with individual EEMs (text data excluded and replaced with 1:N)
% filelist=list of filenames
% outdata = data that were saved to file, or [] otherwise
%%%%%
% Copyright (C) KR Murphy, 2010,
% The University of New South Wales
% krm@unsw.edu.au
%%%%%

```

```

%% INITIALISE
direc = dir(['*.' format]);
filelist=[]; X=[];Emmat=[];Exmat=[]; x=[]; outdata=[];
EmDummy=0;ExDummy=0;

for i=1:size(direc,1)

fprintf([direc(i).name '\n'])
filelist=char(filelist,direc(i).name); %file list

%% LOAD FILES, excluding cells that contain text
if strcmp(format,'csv')==1;
    x = dlmread(direc(i).name,',',range);
elseif strcmp(format,'xls')==1;
    x = xlsread(direc(i).name,'Sheet1',range);
elseif strcmp(format,'xlsx')==1;
    x = xlsread(direc(i).name,'Sheet1',range);
end
if i==1;
    xsize=size(x);
else
    %check that the size of x is consistent with the first EEM
    if size(x)~=xsize
        fprintf('The size of the input EEMs has changed!\n')
        error('Check raw data file and resize as necessary.')
    end
end

%% REMOVE HEADERS
%size(x),x(1:5,1:5)
if headers(2)==1; %remove the Em headers
    if headers(1)==0 %no Ex headers
        Emmat(:,i)=x(:,1);
        if type==1;
            x=x(:,2:end); % remove Em headers from first column
        elseif type==2;
            x=x(:,2:2:end); % remove Em headers from every second column of varian file
        end
        ExDummy=1;Exmat(i,:)=1:size(x,2);
    elseif headers(1)==1 %has Ex headers
        Emmat(:,i)=x(2:end,1);
        if type==1;
            Exmat(i,:)=x(1,2:end);
            x=x(2:end,2:end); % remove Em headers from first column
        elseif type==2;
            Exmat(i,:)=x(1,2:2:end);
            x=x(2:end,2:2:end); % remove Em headers from every second column of varian file
        end
    end
end

```

```

elseif headers(2)==0; % no Em headers
    if type==2
        error('type=2 (alternate columns of Em) is not compatible with "no Em headers"');
    end
    EmDummy=1;
    if headers(1)==1; %remove the Ex headers
        Exmat(i,:)=x(1,:);
        x=x(2:end,:);
    elseif headers(1)==0; %
        ExDummy=1;
        Exmat(i,:)=[1:(size(x,2))];
    end
    Emmat(:,i)=[1:size(x,1)]';
end

X(i,:,:)=x;
%size(Exmat),size(Emmat), size(x),x(1:5,1:5)

if ~isequal(size(Emmat,1),size(x,1))
    fprintf('Em size is '), size(Emmat),
    fprintf('X size is '), size(x),
    error('Em and X matrix sizes not compatible'),
end
if ~isequal(size(Exmat,2),size(x,2))
    fprintf('Ex size is '), size(Exmat),
    fprintf('X size is '), size(x),
    error('Ex and X matrix sizes not compatible'),
end

%% DISPLAY a plot of each EEM as it is loaded
if display==1;
    %      size(x),size(Emmat(:,i)), size(Exmat(i,:))
    figure, contourf(Exmat(i,:),Emmat(:,i),x),
    title(direc(i).name)
    pause(1),
    close
end
end

%% EXPORT DATA
% Export EEM cube to MATLAB file "RawData_FDOM.mat"
% Export data for particular wavelength pairs to an excel spreadsheet "OutData_FDOM.xls"
% adjust or add to the following wavelength selection as required

if outdat==1;
    outwaves=[ 350 450; ...      %C (humic-like)
               250 450; ...      %A (humic-like)
               290 350; ...      %T (tryptophan-like)
               270 304; ...      %B (tyrosine-like)

```

```

320 412]; %M (marine/microbial-like)

%Create outdata matrix (assume Em and Ex as for the first file that was read in)
if EmDummy==0
    Em=round(Emmat(:,1)*2)/2; %round to nearest 0.5 nm
elseif EmDummy==1
    moveon=0;
    while moveon==0;
        Em = input('Specify the emission wavelength range, e.g. 300:2:600: ');
        if size(Em,2)==size(x,1)
            moveon=1;
        else
            fprintf(['Size mismatch: expecting a header of size 1x' num2str(size(x,1)) '. Try again. \n']);
        end
    end
end
if ExDummy==0
    Ex=round(Exmat(1,:)*2)/2; %round to nearest 0.5 nm
elseif ExDummy==1
    moveon=0;
    while moveon==0;
        Ex = input('Specify the excitation wavelength range, e.g. 250:5:450: ');
        if size(Ex,2)==size(x,2)
            moveon=1;
        else
            fprintf(['Size mismatch: expecting a header of size 1x' num2str(size(x,2)) '. Try again. \n']);
        end
    end
end
outdata=NaN*ones(size(X,1)+2,size(outwaves,1)+1); %first two rows are headers
outdata(1,2:end)=outwaves(:,1)'; %excitation headers
outdata(2,2:end)=outwaves(:,2)'; %emission headers
outdata(3:end,1)=(1:size(X,1))'; %number samples
for i=1:size(outwaves,1)
    p=X(:,Em==outwaves(i,2),Ex==outwaves(i,1));
    if ~isempty(p)
        outdata(3:end,i+1)=p;
    end
end
%Write outdata matrix to excel file: OutData_FDOM.xls
fprintf('writing OutData_FDOM.xls to your current directory...\n')
xlswrite('OutData_FDOM.xls',outdata,'Raw')
xlswrite('OutData_FDOM.xls',cellstr('Ex wave'), 'Raw', 'A1')
xlswrite('OutData_FDOM.xls',cellstr('Em wave'), 'Raw', 'A2')
xlswrite('OutData_FDOM.xls',(1:size(X,1))', 'Filenames', 'A2')
xlswrite('OutData_FDOM.xls',cellstr(filelist), 'Filenames', 'B1')
xlswrite('OutData_FDOM.xls',cellstr('List of Files'), 'Filenames', 'B1')

```

```
fprintf('writing RawData_FDOM.mat to your current directory...\n')
save RawData_FDOM.mat X Emmat Exmat filelist outdata
end
```

K2: MATLAB implementation of EEM correction algorithms

The MATLAB function " FDOMcorrect.m" was developed to facilitate fast, accurate and flexible implementation of the EEM correction algorithms presented in this paper. Future revisions or improvements related to this script will be posted on the internet at: <http://www.models.life.ku.dk/algorithms>

Copy and paste the entire code into a single M-file named FDOMcorrect and place it in the MATLAB working directory. Confirm that the file path can be located from the current directory using:

```
>which FDOMcorrect
```

Instructions for using the M-file are obtained by typing:

```
>help FDOMcorrect
```

Examples

The simplest correction method requires 6 inputs and 2 outputs to produce spectrally corrected EEMs in RU units. By default, Raman areas are calculated from the water Raman scan at $\lambda_{ex} = 350$, integrated over $\lambda_{em} = 381-426$ nm. This method excludes inner filter correction, blank subtraction and QS calibration:

```
> [Xc Arp] = FDOMcorrect(X, Ex, Em, Emcor, Excor, W);
```

If performing additional correction steps, all 12 input variables and 6 output variables must be specified and irrelevant variables appearing on the left hand side will be outputted as NaNs. For example:

Inner filter correction and QS calibration (BcRU outputted as NaN):

```
> [XcRU Arp IFCmat BcRU XcQS QS_RU] = FDOMcorrect(X, Ex, Em, Emcor, Excor, W, [], A, [], [], Q, Qw);
```

Inner filter correction and blank subtraction (XcQS, QS_RU outputted as NaN):

```
> [XcRU Arp IFCmat BcRU XcQS QS_RU] = FDOMcorrect(X, Ex, Em, Emcor, Excor, W, [], A, B, [], [], []);
```

Blank subtraction and QS calibration (IFCmat outputted as NaN):

```
> [XcRU Arp IFCmat BcRU XcQS QS_RU] = FDOMcorrect(X, Ex, Em, Emcor, Excor, W, [], [], B, [], Q, Qw);
```

If normalizing to user-specified parameters for Raman area, all 12 input variables and 6 output variables must be specified and irrelevant variables appearing on the left hand side will be outputted as NaNs. For example, to normalize fluorescence intensities to the *height* of the Raman peak at 275/303 nm and perform QS calibration (IFCmat, BcRU outputted as NaN):

```
> [XcRU Arp IFCmat BcRU XcQS QS_RU] = FDOMcorrect(X, Ex, Em, Emcor, Excor, W, [275 303 303], [], [], [], Q, Qw);
```

Troubleshooting

Method statements outputted to screen and saved to file document the program's progress.

Error messages alert the user to incorrectly formatted input files or to incompatibilities between input data and the function request statement. In the final example above using Raman *height* normalization, if $Em = 303$ nm is not among the measurement wavelengths in each of the inputted Raman files (in this case, W and Q_w), the program will abort by error. When performing Raman *area* normalization, however, if one or more Raman files do not include the exact wavelengths requested as upper and lower emission wavelengths, the program will automatically seek an alternative wavelength range, print it to screen, and continue processing. Note that if this occurs, it is advisable to run the program again specifying an emission range that is suitable for all of the input Raman files, so that Raman areas are calculated consistently across the entire dataset.

```

function varargout=FDOMcorrect(X,Ex,Em,Emcor,Excor,W,varargin)
%function [XcRU Arp IFCmat BcRU XcQS QS_RU]=FDOMcorrect(X,Ex,Em,Emcor,Excor,W,RamOpt,A,B,T,Q,Qw);
%
% INPUT VARIABLES:
% Matrix sizes are indicated by (Rows x Columns) in 2D case or (Samples x Rows x Columns) in 3D case.
%
% Ex -1D row vector of excitation wavelengths corresponding to EEMs.
% Em -1D row vector of emission wavelengths corresponding to EEMs.
% X -3D data cube (NO HEADERS, i.e. samples x Em x Ex) of fluorescence intensities in primary samples.
% Emcor -2D matrix (wavelength x correction factor); first column is wavelength.
% Excor -2D matrix (wavelength x correction factor); first column is wavelength.
% W - is either of (a) or (b):
%     (a) a single water Raman emission scan (Ex==landa) applied to all samples; first row is wavelength;
%         (b) 2D matrix (samples x Em) of water Raman scans (Ex==landa) for corresponding samples; first row is wavelength.
% RamOpt - options (user-defined excitation and emission wavelengths) for calculating Raman areas
%           in the form: [landa Em1 Em2]. If RamOpt=[], the default values [350 381 426] are used.
%           RamOpt(1)= landa; excitation wavelength corresponding to the emission scan used for calculating Raman areas.
%           RamOpt(2)= first emission wavelength in the integration range
%           RamOpt(3)= last emission wavelength in the integration range
%
% IF performing INNER FILTER CORRECTION using the Absorbance method, otherwise A=[]:
% A -2D matrix (samples x wavelengths) of Absorbances (decadal form, 1 cm cell); first row is wavelength.
%
% IF performing BLANK SUBTRACTION, otherwise B=[],T=[]:
% B - is either of (a) or (b):
%     (a) single blank EEM (Em x Ex) to be used for all samples;
%         (b) 3D data cube (samples x Em x Ex) of fluorescence intensities in blanks matched with samples.
% T - is either of (a) or (b):
%     (a) T=[] indicating that T is to be automatically extracted from B at Ex=landa;
%         (b) 2D matrix (samples x Em) of water Raman scans(Ex==landa) for blanks; first row is wavelength.
%
% IF performing QUININE SULFATE (QS) CALIBRATION, otherwise Q=[],Qw=[]:
% Q - is either of (a) or (b):
%     (a) a scalar representing the slope (uncorrected) of a linear QS dilution series at 350/450 nm
%         - applied to all samples in the dataset;
%     (b) a vector of slopes (uncorrected) of QS dilution series corresponding with each sample.
% Qw - is either of (a) or (b):
%     (a) a single water Raman emission scan (Ex==landa) for the QS dilution series; first row is wavelength;
%         (b) 2D matrix (samples x Em) of water Raman scans(Ex==landa) corresponding to the vector of QS blanks; first row is wavelength.
%
%% OUTPUT VARIABLES
% XcRU - 3D matrix of corrected EEMs in Raman units, including inner filter correction and blank subtraction steps, if applied.
% Arp - vector of Raman Areas in arbitrary units.
% IFCmat -3D matrix of inner filter correction factors.
% BcRU -3D matrix of corrected blanks.
% XcQS - 3D matrix of corrected EEMs in QS units, including inner filter correction and blank subtraction steps, if applied.
% QS_RU - vector of conversion factors between QS and RU calibrated EEMs.
%
```

```

%% EXAMPLES
% The simplest correction method requires 6 inputs and 2 outputs to produce spectrally corrected EEMs in RU units
% There are no inner filter correction, blank subtraction or QS calibration steps:
% [Xc_Arp]=FDOMcorrect(X,Ex,Em,Emcor,Exc,Cor,W); %spectral correction in RU units with landa = 350 nm
%
% if performing additional or non-default correction steps, all 12 input variables and 6 output variables must be specified
% and irrelevant variables will be outputted as NaN. For example:
% [XcRU_Arp IFCmat BcRU_XcQS_QS_RU]=FDOMcorrect(X,Ex,Em,Emcor,Exc,Cor,W,[],A,[],[],[],[]); %inner filter correction
% [XcRU_Arp IFCmat BcRU_XcQS_QS_RU]=FDOMcorrect(X,Ex,Em,Emcor,Exc,Cor,W,[],A,B,[],[],[]); %inner filter correction, blank subtraction
% [XcRU_Arp IFCmat BcRU_XcQS_QS_RU]=FDOMcorrect(X,Ex,Em,Emcor,Exc,Cor,W,[],[],B,[],Q,Qw); %blank subtraction, RU and QS normalization
% [XcRU_Arp IFCmat BcRU_XcQS_QS_RU]=FDOMcorrect(X,Ex,Em,Emcor,Exc,Cor,W,[280 300 325],[],[],[],Q,Qw); %RU, QSE units with landa = 280 nm
%
% When calling the function, use ";" at the end of the function call to prevent the output variables
% from being printed to screen, as seen in the above examples.
%
%% ABOUT THIS MATLAB FILE
% These algorithms are a formalisation of the EEM correction procedures resulting from discussions at the
% AGU Chapman Conference on organic matter fluorescence, held in Birmingham UK in October, 2008.
% Please cite this program as: FDOMcorrect.m
% in Murphy et al. 2010 'The measurement of dissolved organic matter fluorescence in aquatic environments: An inter-laboratory comparison'
% Copyright (C) 2010 KR Murphy,
% The University of New South Wales
% Department of Civil and Environmental Engineering
% Sydney 2052, Australia
% krm@unsw.edu.au
%%
format bank
error(nargchk(6, 12, nargin))

%Set up File Export preferences with ExportFiles=true or ExportFiles=false;
%If ExportFiles==true, the following data are saved to file in the current directory.
% (a) "OutData_FDOM.xls": corrected intensities at common wavelength pairs (A,C,M,T,B)
% (b) "CorrData_FDOM.mat": XcRU,Arc,X,Ex,Em,Emcor,Exc,Cor are saved to a MATLAB file
ExportFiles=true; %true/false

%Default methods
IFEcorrection=false;
QScalibration=false;
BlankSubtraction=false;
method1='METHOD: Perform spectral correction';
method2='METHOD: Perform RU normalisation';

%inputs
if nargin==6;
    error(nargoutchk(2,2,nargout))
    method3='METHOD: No inner filter correction';
    method4='METHOD: No blank subtraction';
    method5='METHOD: No QS calibration';

```

```

IFCmat=NaN*ones(size(X));
BcRU=NaN*ones(size(X));
XcQS=NaN*ones(size(X));
methodopts=char(['\n',method1,'\n',method2,'\n',method3,'\n',method4,'\n',method5,'\n']);
else
if nargin<12;
    error('function requires either 6 or 12 input variables.')
elseif nargin==12;
    RamOpt=varargin{1};
    A=varargin{2};
    B=varargin{3};
    T=varargin{4};
    Q=varargin{5};
    Qw=varargin{6};
    error(nargoutchk(6,6,nargout))
end

if isempty(RamOpt),
    landa=350; Em1=381; Em2=426;
else
    landa=RamOpt(1); Em1=RamOpt(2); Em2=RamOpt(3);
end
method3=[ 'METHOD: Raman areas calculated for Ex = ' num2str(landa) ' nm'];
method4=[ 'METHOD: Raman areas to be integrated over Em = ' num2str(Em1) ' to ' num2str(Em2) ' nm'];

if ~isempty(A);
    IFEcorrection=true;
    method5='METHOD: Perform inner filter correction';
else
    IFEcorrection=false;
    method5='METHOD: No inner filter correction';
    IFCmat=NaN*ones(size(X));
end

if ~isempty(B);
    BlankSubtraction=true;
    method6='METHOD: Perform blank subtraction';
else
    BlankSubtraction=false;
    method6='METHOD: No blank subtraction';
    BcRU=NaN*ones(size(X));
end

if and(~isempty(Q),~isempty(Qw))
    QScalibration=true;
    method7='METHOD: Perform QS calibration';
elseif and(isempty(Q),isempty(Qw))
    QScalibration=false;
    method7='METHOD: No QS calibration';

```

```

XcQS=NaN*ones(size(X));
else
    error('QS calibration requires both slope and Raman inputs. Wrong number of input arguments!');
end
methodopts=char('\n',method1,'\n',method2,'\n',method3,'\n',method4,'\n',method5,'\n',method6,'\n',method7,'\n');
end
fprintf(methodopts')

%% Step 0: Pretreatment: Resize files as needed
fprintf('\n\n')
fprintf('Step 0: Checking input files...\n')
N_samples=size(X,1); %or: N_samples=max(size(L));
N_ex=max(size(Ex));
N_em=max(size(Em));
Em=CheckMatrixDim(Em,N_em,1,[],'Em');
Ex=CheckMatrixDim(Ex,1,N_ex,[],'Ex');
X=CheckMatrixDim(X,N_samples,N_em,N_ex,'X');

% Obtain water Raman Scans for Samples
fprintf(' obtaining water Raman scans, W...\n')
if min(size(W))==2; % use a single Raman scan for all samples
    W=CheckMatrixDim(W,2,[],[],'W');
    W=[W(1,:)' W(2,:)'*ones(1,N_samples)]';
end
W=CheckMatrixDim(W,N_samples+1,[],[],'W');

%Obtain correction files and resize as necessary
doplots=false;
fprintf(' obtaining and resizing correction files...\n')
fprintf(' matching EmcorX & Em \n');
EmcorX=MatchWave(Emcor,Em,doplots); %Resize EmcorX to match Em,
fprintf(' matching ExcorX & Ex \n');
ExcorX=MatchWave(Excor,Ex,doplots); %Resize ExcorX to match Ex,
fprintf(' matching Emcor & W \n');
EmcorR=MatchWave(Emcor,W,doplots); %Resize Emcor to match Raman scans for samples
EmcorX=CheckMatrixDim(EmcorX,N_em,2,[],'Emcor for EEM');
ExcorX=CheckMatrixDim(ExcorX,N_ex,2,[],'Excor for EEM');
EmcorR=CheckMatrixDim(EmcorR,size(W,2),2,[],'Emcor for Raman Scan');
fprintf(' obtaining correction factors @ 350/450 nm \n');
Excor350=Excor(Excor(:,1)==350,2);
Emcor450=Emcor(Emcor(:,1)==450,2);
cor350_450=Excor350*Emcor450;
if ~isempty(cor350_450)
else error('Could not locate ex=350 and em=450 nm in correction files');
end
Excorlanda=Excor(Excor(:,1)==landa,2);
if ~isempty(Excorlanda)
else error('Could not locate ex=landa in correction files');
end

```

```

if BlankSubtraction==true;
    fprintf(' obtaining and resizing blank files, B and T...\n')
    %Check size of B and replicate single EEM if necessary
    B=squeeze(B);
    if ndims(B)==2; %2D dataset
        fprintf(' replicating blank EEM .. \n');
        B=CheckMatrixDim(B,N_em,N_ex,[],'B');
        B=RepEEM(B,N_samples,0);
    end
    B=CheckMatrixDim(B,N_samples,N_em,N_ex,'B');

    %Check size of T and extract from B if necessary
    if isempty(T);
        fprintf(' Raman scans for Blanks will be extracted from blank matrix\n');
        T=[Em';squeeze(B(:,:,Ex==landa))];
    end
    T=CheckMatrixDim(T,N_samples+1,[],[],'T');
end

if QScalibration==true;
    fprintf(' obtaining and resizing dilution series Qs...\n')
    % Q is a scalar or vector of dilution series slopes
    if max(size(Q))==1; %a scalar
        fprintf(' use same QS slope for all samples .. \n');
        Q=Q*ones(N_samples,1);
    end
    Q=CheckMatrixDim(Q,N_samples,1,[],'Q');

    fprintf(' obtaining and resizing water Raman Qw...\n')
    if min(size(Qw))==2; % a single Raman scan for all samples
        Qw=CheckMatrixDim(Qw,2,[],[],'W');
        Qw=[Qw(1,:)' Qw(2,:)'*ones(1,N_samples)]';
    end
    Qw=CheckMatrixDim(Qw,N_samples+1,[],[],'Qw');
end

%% Step 1: Apply Spectral Correction files
fprintf('\nStep 1: Apply Spectral Correction \n')

%EEM correction factors
eemXcor=EmcorX(:,2)*ExcOrX(:,2)'; %2D correction matrix applying to each sample
eemcormat=RepEEM(eemXcor,N_samples,0); %3D correction matrix

%correct the sample EEMs
Xc = eemcormat.*X;
Xs=Xc; %spectrally corrected EEMs

%correct the Raman scans corresponding to the sample EEMs
[mw ~]=size(W);

```

```

Rcormat=(Excorlanda*EmcorR(:,2)*ones(1, mw))';
Wc=Rcormat.*W;
Wc(1,:)=W(1,:); %Corrected Raman scans, first row is wavelength

if BlankSubtraction==true;
    %Spectrally correct B and T
    Bc = eemcormat.*B; %spectrally corrected blanks
    fprintf(' matching EmcorT & T \n');
    EmcorT=MatchWave(Emcor,T,doplots); %Resize Emcor to match Raman scans for blanks
    [mw ~]=size(T);
    Tcormat=(Excorlanda*EmcorT(:,2)*ones(1, mw))';
    Tc=Tcormat.*T;
    Tc(1,:)=T(1,:); %Tc=spectrally corrected Raman scans for blanks, first row is wavelength
end

if QScalibration==true;
    % Spectrally correct the Quinine Sulfate dilution series and Raman scans
    % Slopes:
    Sc=cor350_450*Q; %Corrected slopes of the dilution series
    % Raman scans:
    fprintf(' matching EmcorQ & Qw \n');
    EmcorQ=MatchWave(Emcor,Qw,doplots);
    [mw ~]=size(Qw);
    Qcormat=(Excorlanda*EmcorQ(:,2)*ones(1, mw))';
    Qwc=Qcormat.*Qw;
    Qwc(1,:)=Qw(1,:); %Qwc=spectrally corrected Raman scans for QS series
end

%% Step 2: Inner Filter Correction (Parker & Barnes 1957)
%A is the absorbance in a 1 cm cell in decadal form (samples x wavelengths)
%method is approximately valid for a254<0.3
%First, resize A to match wavelengths Ex and Em in EEMs

if IFEcorrection==true;
    fprintf('Step 2: Apply Inner Filter correction \n');
    A=CheckMatrixDim(A,N_samples+1,[],[],'A');
    A=(sortrows(A',1))'; %wavelengths in descending order
    if min(A)<0
        warning('2.1: some Absorbance measurements have negative values.');
        warning('2.2: Negative absorbance will be set to zero.');
        A(A<0)=0;
    end

    %Generate a matrix of inner filter corection factors
    %according to the method of Parker & Barnes, 1957.
    pathlength = 1; %1 cm cell
    warning(['2.3: Pathlength of ' num2str(pathlength) '-cm is assumed.']);
    IFCmat = IFEabs(A,N_samples,Em,Ex,pathlength);
    % % OPTIONAL: Plot the correction factors

```

```

% for i=1:size(M,1)
%     figure, contourf(Ex,Em,squeeze(IFCmat(i,:,:)));
% end
Xife =Xc.*IFCmat;
Xc=Xife;
else fprintf('Step 2: No inner filter correction \n')
end

%% Step 3: Raman Normalisation
%Integrate scatter peaks between Em1-Em2 in a matrix of Raman scans
%Em1 and Em2 are scalars to nearest 0.5nm, e.g. 381, 426 nm
%Note that Em1 and Em2 must both be present in the Raman scan.
%If Em1 or Em2 are not found the program will increment Em1 down and Em2 up by 0.5m.
%If Em1 or Em2 are still not found following 10 increments an error will be produced
%and a new integration range must be specified manually.

fprintf('Step 3: Apply Raman normalisation \n')

%Normalise the sample EEMs to the area under the Raman scan between Em1-Em2;
fprintf(' Normalise the EEMs using Raman scans from W. \n')
fprintf(' First, check wavelengths in W and adjust integration range if necessary.... \n')
if Em1==Em2
    Arp=Wc(2:end,Wc(1,:)==Em1);
    if max(isnan(Arp))==1 %NaN in one or more rows
        error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
    elseif isempty(Arp), %all rows empty
        error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
    end
    method2a=[ 'METHOD: Samples normalized to Raman height at' num2str(Em1) ' nm'];
else
    [Arp EmAout1 EmAout2]=RamanArea(Wc,Em1,Em2);
    method2a=[ 'METHOD: Actual Raman integration area - samples: ' num2str(EmAout1) ' to ' num2str(EmAout2) ' nm'];
end
methodopts=char(methodopts,method2a, '\n');

ArpMat=repmat(Arp,[1,N_em,N_ex]); %convert to a 3D matrix
Xc = Xc./ArpMat; %EEMs in R.U.

if BlankSubtraction==true;
    fprintf(' Normalise the blanks using Raman scans from T (or B)... \n')
    fprintf(' First, check wavelengths in T (or B) and adjust integration range if necessary... \n')
    %calculate Raman area/height for the blanks
    if Em1==Em2
        Brp=Tc(2:end,Tc(1,:)==Em1);
        if max(isnan(Brp))==1 %NaN in one or more rows
            error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
        elseif isempty(Brp), %all rows empty
            error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
    end
end

```

```

    end
method2b=[ 'METHOD: Blanks normalized to Raman height at' num2str(Em1) ' nm'];
else
    [Brp EmBout1 EmBout2]=RamanArea(Tc,Em1,Em2);
method2b=[ 'METHOD: Actual Raman integration area - blanks: ' num2str(EmBout1) ' to ' num2str(EmBout2) ' nm'];
end
methodopts=char(methodopts,method2b,'\\n');

BrpMat=repmat(Brp,[1,N_em,N_ex]); %convert to a 3D matrix

%Normalise the blanks, using the Raman area/height corresponding to the blanks
BcRU = Bc./BrpMat; %blanks in R.U.
end

if QS_calibration==true;
    fprintf(' Normalise the QS series using Raman scans from Qw... \\n')
    fprintf(' First, check wavelengths in Qw and adjust integration range if necessary... \\n')
    %Normalise the QS dilution series to Raman area/height.
    if Em1==Em2
        Qr=Qwc(2:end,Qwc(1,:)==Em1);
        if max(isnan(Qr))==1 %NaN in one or more rows
            error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
        elseif isempty(Qr), %all rows empty
            error(['Emission wavelength ' num2str(Em1) '-nm not found in one or more Raman scans.'])
        end
        method2c=[ 'METHOD: QS series normalized to Raman height at' num2str(Em1) ' nm'];
    else
        [Qr EmQout1 EmQout2]=RamanArea(Qwc,Em1,Em2); %Qr=Raman Area for corrected QS blanks (samples x 1)
        method2c=[ 'METHOD: Actual Raman integration area - QS: ' num2str(EmQout1) ' to ' num2str(EmQout2) ' nm'];
    end
    RU_QS=Sc./Qr;
    QS_RU=Qr./Sc;
    QS_RUMat=repmat(QS_RU,[1,N_em,N_ex]); %convert to a 3D matrix
    methodopts=char(methodopts,method2c,'\\n');
end

%% Step 4: Blank Subtraction
if BlankSubtraction==true;
    % blank subtraction step
    fprintf('Step 4: Apply blank subtraction \\n')
    Xbs = Xc - BcRU;
    Xc=Xbs;
else
    fprintf('Step 4: No blank subtraction \\n')
end
XcRU=Xc;

%% Step 5: Quinine Sulfate Calibration
if QS_calibration==true;
    % Calibrate to corrected slope of the dilution series

```

```

fprintf('Step 5: Apply QS calibration \n')
XcQS=Xc.*QS_RUMat; %EEMs in QSE units
% Display: Raw, Corrected, Ram Area, QS/RU
QW350_450=[Q Sc Qr QS_RU RU_QS];
if min(Q(1)*ones(N_samples,1)==Q)==1 %QW350_450 is identical for all samples
    QW350_450=QW350_450(1,:);
end
fprintf(' QS slope at 350/450 nm: \n')
disp('          Raw           Corr.           RamArea           QS/RU           RU/QS ')
disp('          -----           -----           -----           -----           -----')
disp(QW350_450);
else fprintf('Step 5: No QS calibration \n')
    QS_RU=NaN;
end

%% Demonstrate results at specific wavelengths (Ex/Em=pex/pem)
pex=350;
pem=450;

XsRA=Xs(:,(Em==pem), (Ex==pex))./Arp;

p=[X(:,(Em==pem), (Ex==pex)) ...
    Xs(:,(Em==pem), (Ex==pex)) ...
    Arp ...
    XsRA ...
    IFCmat(:,Em==pem,Ex==pex) ...
    BcRU(:,(Em==pem), (Ex==pex)) ...
    XcRU(:,(Em==pem), (Ex==pex))...
    XcQS(:,(Em==pem), (Ex==pex))];
fprintf('Demonstration of the results: ');
fprintf(['\n Sample intensity at ' num2str(pex) '/' num2str(pem) 'nm: ' ...
    '\n Col 1: raw,' ...
    '\n Col 2: corrected,' ...
    '\n Col 3: Raman Area,' ...
    '\n Col 4: spectrally corrected in RU units [default]' ...
    '\n Col 5: inner filter correction factor,' ...
    '\n Col 6: corrected blank,' ...
    '\n Col 7: final output, RU units ' ...
    '\n Col 8: final output, QSE units\n\n'])
disp(p)

```

%% OUTPUT VARIABLES

```

%Convert irrelevant output matrices to simple NaNs
if IFEcorrection==false;IFCmat=NaN;end
if BlankSubtraction==false;BcRU=NaN;end
if QScalibration==false;XcQS=NaN;QS_RU=NaN;end

%output variables

```

```

varargout{1}=XcRU;
varargout{2}=Arp;
if sum([IFEcorrection,BlankSubtraction,QScalibration])>0;
    varargout{3}=IFCmat;
    varargout{4}=BcRU;
    varargout{5}=XcQS;
    varargout{6}=QS_RU;
end

%% EXPORT DATA for particular wavelength pairs to file
% If ExportFiles = 1, data are saved to the current directory:
% (a) "OutData_FDOM.xls": corrected intensities at common wavelength pairs (A,C,M,T,B)
% (b) "CorrData_FDOM.mat": XcRU,Arp,X,Ex,Em,Emcor,Exc or are saved to a MATLAB file

if ExportFiles==1;

    %Adjust or add to the following wavelength selection as required
    outwaves=[350 450; ...      %C (humic-like)
              250 450; ...      %A (humic-like)
              290 350; ...      %T (tryptophan-like)
              270 304; ...      %B (tyrosine-like)
              320 412];         %M (marine/microbial-like)

    %Create OutData matrix
    outdata=NaN*ones(size(XcRU,1)+2,size(outwaves,1)+1); %first two rows are headers
    outdata(1,2:end)=outwaves(:,1)'; %excitation headers
    outdata(2,2:end)=outwaves(:,2)'; %emission headers
    outdata(3:end,1)=(1:size(XcRU,1))'; %number samples
    for i=1:size(outwaves,1)
        p=XcRU(:,Em==outwaves(i,2),Ex==outwaves(i,1));
        if ~isempty(p)
            outdata(3:end,i+1)=p;
        end
    end

    %Write OutData to Excel: "OutData_FDOM.xls":
    fprintf('writing summary data OutData_FDOM.xls to your current directory...\\n')
    xlswrite('OutData_FDOM.xls',cellstr(methodopts(2:2:end,:)), 'Methods')
    xlswrite('OutData_FDOM.xls',outdata, 'Corr_RU')
    xlswrite('OutData_FDOM.xls',cellstr('Ex wave'), 'Corr_RU', 'A1')
    xlswrite('OutData_FDOM.xls',cellstr('Em wave'), 'Corr_RU', 'A2')

    if QScalibration==true
        for i=1:size(outwaves,1)
            outdata(3:end,i+1)=XcQS(:,Em==outwaves(i,2),Ex==outwaves(i,1));
        end
        xlswrite('OutData_FDOM.xls',cellstr('Sample'), 'QS_RU', 'A1')
        xlswrite('OutData_FDOM.xls',cellstr('QS/RU'), 'QS_RU', 'B1')
        xlswrite('OutData_FDOM.xls',[1:size(QS_RU,1)]' QS_RU, 'QS_RU', 'A2')
    end

```

```

xlswrite('OutData_FDOM.xls',outdata,'Corr_QS')
xlswrite('OutData_FDOM.xls',cellstr('Ex wave'), 'Corr_QS','A1')
xlswrite('OutData_FDOM.xls',cellstr('Em wave'), 'Corr_QS','A2')
end
fprintf('                                .....done. \n')

%Write the full list of variables to "CorrData_FDOM.mat":
fprintf('writing CorrData_FDOM.mat to your current directory... \n')
if sum([IFEcorrection,BlankSubtraction,QScalibration])>0;
    save CorrData_FDOM.mat XcRU Arp IFCmat BcRU XcQS QS_RU X Ex Em Emcor Exc or W RamOpt A B T Q Qw
else
    save CorrData_FDOM.mat XcRU Arp X Ex Em Emcor Exc or W
end
end %Export Files

%% NESTED FUNCTIONS
function M=CheckMatrixDim(M,Dim1,Dim2,Dim3,Mname)
% Check the dimensions of matrix M (name = 'Mname') against expected dimensions Dim1, Dim2, and Dim3
% Transpose M if necessary to achieve correct dimensions or else generate error message.
% Copyright K.R. Murphy,
% July 2010
if isempty(Dim3); %2D data
    if size(M,1)~=Dim1;
        if and(size(M,2)==Dim1,size(M,1)==Dim2)
            M=M';
        elseif isempty(Dim2) %Number of columns is arbitrary
            if size(M,2)==Dim1
                M=M';
            else
                fprintf(['Check size of matrix ' Mname '\n'])
                fprintf(['Expecting ' num2str(Dim1) ' rows.\n']),pause
                fprintf(['Current size is ' num2str(size(M,1)) ' rows and ' num2str(size(M,2)) ' columns.\n']),pause
                fprintf('Hit any key to continue.\n'),pause
                error('Unexpected Matrix Size')
            end
        else
            fprintf(['Check size of matrix ' Mname '\n'])
            fprintf(['Expecting ' num2str(Dim1) ' rows and ' num2str(Dim2) ' columns.\n']),pause
            fprintf(['Current size is ' num2str(size(M,1)) ' rows and ' num2str(size(M,2)) ' columns.\n']),pause
            fprintf('Hit any key to continue.\n'),pause
            error('Unexpected Matrix Size')
        end
    end
else %3D data
    if isequal(size(M),[Dim1 Dim2 Dim3]);
    else
        fprintf(['Check size of matrix ' Mname '\n'])
        fprintf(['Expecting ' num2str(Dim1) ' x ' num2str(Dim2) ' x ' num2str(Dim3) '\n']),pause
        fprintf(['Current size is ' num2str(size(M,1)) ' x ' num2str(size(M,2)) ' x ' num2str(size(M,3)) '\n']),pause
    end
end

```

```

fprintf('Hit any key to continue.\n'),pause
error('Unexpected Matrix Size')
end
end

%%
function Y=RepEEM(X,N,dochecks)
% Make an cube of identical EEMs, each consisting of the EEM X
% Size of cube specified by input N
% if dochecks = true, function displays 3 matrices which should all be identical.
% Copyright K.R. Murphy,
% July 2010

n1=size(X,1);
n2=size(X,2);
X1 = repmat(X,N,1);
X1 = reshape(X1,[n1 N n2]);
Y = permute(X1,[2 1 3]);

if dochecks==true;
    squeeze(X(1:5,1:3))
    squeeze(Y(1,1:5,1:3))
    squeeze(Y(2,1:5,1:3))
    fprintf('check that the preceeding 3 matrices are all identical, then press any key to continue...')
    pause
end

%%
function Y=MatchWave(V1,V2,doplot)
%Automatically match the size of two vectors, interpolating if necessary.
%Vector V1 is resized to have the same wavelengths as V2;
%For example, V1 is a correction file (0.5nm intervals) and V2 is an Emission scan in 2 nm intervals.
%Errors are produced if
%(1) there are wavelengths in the emission scan that have no correction factors.
%(2) the first or last wavelength in the emission scan does not have an exactly corresponding correction factor.
% Errors can often be resolved by restricting the wavelength range of V2
% Copyright K.R. Murphy
% July 2010

if size(V1,2)>2; V1=V1';end %V1 should have multiple rows
if size(V2,2)>2; V2=V2';end %V2 should have multiple rows

t=V1(find(V1(:,1)==V2(1,1)):find(V1(:,1)==V2(end,1)),:); %Restrict wavelength range of V1 to same as V2
if isempty(t)
    fprintf(['Error - Check that your VECTOR 1 (e.g. correction file) ...
    'encompasses \n the full range of wavelengths in VECTOR 2 (e.g. emission scan)\n'])
    fprintf('also, the very first and last wavelengths in VECTOR 2 need to appear in VECTOR 1. \n Hit any key to continue...')
    pause
    fprintf(['\n VECTOR 1 range is ' num2str(V1(1,1)) ' to ' num2str(V1(end,1)) ' in increments of ' num2str(V1(2,1)-V1(1,1)) '.']),pause

```

```

fprintf(['\n VECTOR 2 range is ' num2str(V2(1,1)) ' to ' num2str(V2(end,1)) ' in increments of ' num2str(V2(2,1)-V2(1,1)) '.\n'],pause
fprintf('\n This error can often be resolved by restricting the wavelength range of VECTOR 2.\n');
error('Error - abandoning calculations.');
else
    Y=[V2(:,1) interp1(t(:,1),t(:,2),V2(:,1))]; %V1 corresponding with EEM wavelengths
end

if doplot==true;
figure,
plot(V1(:,1),V1(:,2)), hold on, plot(Y(:,1),Y(:,2), 'ro')
legend('VECTOR 1', 'VECTOR 2')
end

function M=IFEabs(A,N_samples,Em,Ex,PL)
%IFEmat = IFEabs(A,N_samples,Em,Ex,PL)
%Calculate a matrix of inner filter correction factors according to the
%absorbance method of Parker & Barnes (1957).
%Parker, C. A.; Barnes, W. J., Some experiments with spectrofluorimeters
%and filter fluorimeters. Analyst 1957, 82, 606-618.
%Inputs:
%A = 2D matrix of Absorbance scans with wavelength on the first row
%N_samples=number of samples
%Em=vector of emission wavelengths
%Ex=vector of excitation wavelengths
%PL = pathlength (1 cm cell)

N_em=length(Em);
N_ex=length(Ex);
M=NaN*ones([N_samples,N_em,N_ex]);

for k=1:N_samples;
    A254=A(2:end,round(A(1,:))==254);
    if max(A254>0.3)
        warning(['2.4: A254 exceeds 0.3 for one or more samples. ...
                  'The current inner filter correction method is unsuitable for highly concentrated samples. ...
                  'Consider restricting the dataset or using an alternative IFE correction method.']);
        figure,
        subplot(2,1,1),
        plot(A(1,:)',A(2:end,:))
        hold on, plot(A(1,:)',0.05*ones(size(A,2)), 'k--')
        hold on, plot(A(1,:)',0.3 *ones(size(A,2)), 'k--')
        hold on, plot(254*ones(11,1),0:0.1:1, 'k-')
        axis tight,
        ylim([0 1])
        xlabel('wavelength (nm)'), ylabel('Absorbance');
        subplot(2,1,2),
        plot(A254, 'ro')
        xlabel('sample number'), ylabel('Absorbance at 254 nm');
    end

```

```

Atot=zeros(N_em,N_ex);
Ak=[A(1,:);A(k+1,:)];
%fprintf('matching Absorbance scans & Ex, Em \n');
Aem=MatchWave(Ak,Em,0);
Aex=MatchWave(Ak,Ex,0);

for i=1:size(Aem,1)
    for j=1:size(Aex,1)
        Atot(i,j)=Aex(j,2)+Aem(i,2);
    end
end
IFC=10.^((PL/2*Atot));
M(:, :, :)=IFC;
end

function [Y,EmMin,EmMax]=RamanArea(M,EmMin,EmMax)
% function Y=RamanArea(M,EmMin,EmMax)
% Find the area under the curves in M between wavelengths EmMin and EmMax, with EmMin<EmMax
% Wavelengths must be in the first row of M. These are rounded automatically to nearest 0.5nm
% prior to matching with EmMin and EmMax.
% If EmMin or EmMax are not found the program will increment Em1 down and Em2 up by 0.5m for a maximum
% of 10 incrementation steps, after which an error will be produced.
% Fix this error by adjusting EmMin and EmMax or by interpolating M.
% Copyright K.R. Murphy
% July 2010

wave=M(1,:); %first row is wavelength
wave=round(wave*2)/2; %round wavelengths to nearest 0.5 nm
istart=find(wave==EmMin); %index of first wavelength
istop=find(wave==EmMax); %index of last wavelength

j=0; jmax=10;
while isempty(istart),
    j=j+1;
    fprintf(['      Wavelength ' num2str(EmMin) 'nm not found in Raman scan. Automatically adjusting integration range...\n']),
    EmMin = EmMin-0.5;
    istart=find(wave==EmMin); %index of first wavelength
    if j>jmax
        error('Can not find integration wavelengths in Raman file. Check files before continuing.')
    end
end
j=0; jmax=10;
while isempty(istop),
    fprintf(['      Wavelength ' num2str(EmMax) 'nm not found in Raman scan. Automatically adjusting integration range...\n']),
    EmMax = EmMax+0.5;
    istop=find(wave==EmMax); %index of first wavelength
    if j>jmax;
        error('Can not find integration wavelengths in Raman file. Check files before continuing.')
    end
end

```

```

    end
end
fprintf(['    Actual integration range is ' num2str(EmMin) ' to ' num2str(EmMax) ' nm.\n'])
delta=round((EmMax-EmMin)/(istop-istart)*2)/2; %wavelength increment to nearest 0.5 nm

%Area under the curve
rr=M(2:end,istart:istop); %restrict to curve between EmMin and EmMax
rr1=rr(:,1:end-1);
rr2=rr(:,2:end);
RAsum=0.5*(sum(rr1,2)+sum(rr2,2))*delta;

%Calculate area below the baseline
y1=rr(:,1);y2=rr(:,end); %data points at EmMin and EmMax;
BaseArea=(wave(istop)-wave(istart))*(y1+0.5*(y2-y1));

%Raman Area, Y
Y = RAsum - BaseArea;

```