Supporting Information A.

A model for converting soil volumetric water content to water potential

Gwang-Jung Kim

Department of Forest Ecology and Management, Swedish University of Agricultural Sciences (SLU), Umeå, Sweden

To model seed germination using hydrothermal time (HTT) framework, soil water potential (Ψ_{soil}, MPa) was converted from volumetric soil water content (θ, %). The conversion followed the model established by Saxton and Rawls (2006), as follows:

$$\Psi_\theta = \begin{cases} A(\theta)^{-B}, & \theta \le \theta_{33} \\ 33 - \left[\dfrac{(\theta - \theta_{33})(33 - \Psi_e)}{\theta_S - \theta_{33}}\right], & \theta > \theta_{33} \end{cases}$$

Equations for parameters

Coefficients of moisture-tension

$$A = \exp\left(\ln 33 + B \ln \theta_{33}\right)$$

$$B = \frac{[\ln(1500) - \ln(33)]}{[\ln(\theta_{33}) - \ln(\theta_{1500})]}$$

33 kPa moisture

$$\theta_{33} = \theta_{33t} + [1.283(\theta_{33t})^2 - 0.374(\theta_{33t}) - 0.015]$$

$$\theta_{33t} = -0.251S + 0.195C + 0.011OM + 0.006(S \times OM) - 0.027(C \times OM) + 0.452(S \times C) + 0.299$$

Tension at air entry

$$\Psi_e = \Psi_{et} + (0.02\Psi_{et} - 0.113\Psi_{et} - 0.70)$$

$$\Psi_{et} = -21.675 - 27.93C + 0.971OM + 71.12S \times (\theta_S - \theta_{33}) + 8.29(C \times (\theta_S - \theta_{33})) + 14.05(S \times C) + 27.16$$

Saturation parameters

$$\theta_S = \theta_{33} + \theta_{(S-33)} - 0.097S + 0.043$$

$$\theta_{(S-33)} = \theta_{(S-33)t} + (0.636\theta_{(S-33)t} - 0.107)$$

$$\theta_{(S-33)t} = 0.278S + 0.034C + 0.022OM + 0.018(S \times OM) + 0.027(C \times OM) + 0.584(S \times C) + 0.078$$

Variables

S: sand (%)

C: clay (%)

OM: organic matter (%). *Note: Since soil organic matter was not directly measured, we assumed an organic matter of 2.5% for this analysis.*

References

Saxton, SE & Rawls, WJ. 2006. Soil water characteristic estimates by texture and organic matter for hydrologic solutions. *Soil Science Society of America Journal*. 70(5): 1569–1578.

Converting soil volumetric water content to water potential with R: code snippets

```
cal_poten <- function(S, C, OM, theta) {

  # Function to calculate theta_33t
  theta_33t <- function(S, C, OM) {
    -0.251 * S + 0.195 * C + 0.011 * OM + 0.006 * (S * OM) - 0.027 * (C *
OM) + 0.452 * (S * C) + 0.299
  }

  # Function to calculate theta_33
  theta_33 <- function(S, C, OM) {
    theta_33t_value <- theta_33t(S, C, OM)
    theta_33t_value + (1.283 * theta_33t_value^2 - 0.374 * theta_33t_value
- 0.015)
  }

  # Function to calculate theta_S_33t
  theta_S_33t <- function(S, C, OM) {
    0.278 * S + 0.034 * C + 0.022 * OM + 0.018 * (S * OM) + 0.027 * (C *
OM) + 0.584 * (S * C) + 0.078
  }

  # Function to calculate theta_s
  theta_s <- function(S, C, OM) {
    theta_33_value <- theta_33(S, C, OM)
    theta_S_33_value <- theta_S_33t(S, C, OM)
    theta_33_value + theta_S_33_value - 0.097 * S + 0.043
  }

  # Function to calculate psi_e
```

```r
  psi_et <- function(S, C, OM, theta_s, theta_33) {
    -21.675 - 27.93 * C + 0.971 * OM + 71.12 * S * (theta_s - theta_33) +
      8.29 * C * (theta_s - theta_33) + 14.05 * S * C + 27.16
  }

  # Function to calculate theta_1500t
  theta_1500t <- function(S, C, OM) {
    -0.024 * S + 0.487 * C + 0.006 * OM + 0.005 * (S * OM) - 0.013 * (C *
OM) + 0.068 * (S * C) + 0.031
  }

  # Function to calculate theta_1500
  theta_1500 <- function(S, C, OM) {
    theta_1500t_value <- theta_1500t(S, C, OM)
    theta_1500t_value + (0.14 * theta_1500t_value - 0.02)
  }

  # Function to calculate B
  calculate_B <- function(theta_33_value, theta_1500_value) {
    (log(1500) - log(33)) / (log(theta_33_value) - log(theta_1500_value))
  }

  # Function to calculate A
  calculate_A <- function(B, theta_33_value) {
    exp(log(33) + B * log(theta_33_value))
  }

  # Calculate intermediate values
  theta_33_value <- theta_33(S, C, OM)
  theta_s_value <- theta_s(S, C, OM)
  theta_1500_value <- theta_1500(S, C, OM)
  psi_e_value <- psi_et(S, C, OM, theta_s_value, theta_33_value)

  # Calculate B and A
  B <- calculate_B(theta_33_value, theta_1500_value)
  A <- calculate_A(B, theta_33_value)

  # Function to calculate psi_0
  psi_0 <- function(theta) {
    if (theta <= theta_33_value) {
      # Case 1: theta <= theta_33
      return(A * theta^(-B))
    } else {
      # Case 2: theta > theta_33
      return(33.0 - ((theta - theta_33_value) * (33.0 - psi_e_value)) /
(theta_s_value - theta_33_value))
    }
  }

  # Return all calculated values as a list
  return(list(
    theta_33 = theta_33_value,
    theta_s = theta_s_value,
    theta_1500 = theta_1500_value,
```

```
    psi_e = psi_e_value,
    B = B,
    A = A,
    psi_0 = psi_0(theta)
  ))
}
```