

Guía de usuario del modelo MiRoya_DynaCof



Contacto: Pierre Bommel (bommel@cirad.fr)

Plano

| | |
|--|-----------|
| AUTORES QUE HAN PARTICIPADO EN EL DISEÑO DEL MODELO | 3 |
| OBJETIVO DE MIROYA | 4 |
| DESCRIPCIÓN DEL MODELO CONCEPTUAL | 6 |
| ESTRUCTURA DEL MODELO | 6 |
| DINÁMICAS DEL MODELO | 8 |
| TIEMPO | 8 |
| CICLO DE VIDA DE LA ROYA | 9 |
| Lixiviación | 11 |
| Contaminación | 11 |
| Germinación | 12 |
| Penetración | 12 |
| Período de latencia y de esporulación | 12 |
| Eспорулación | 13 |
| Duración de la lesión esporulante | 13 |
| Mortalidad por alta temperatura | 13 |
| DINAMICA DEL CAFETO | 13 |
| CAÍDA DE LA HOJA POR LA ROYA (RELACIÓN ESTADÍSTICA) | 14 |
| IMPLEMENTACIÓN DEL MODELO | 15 |
| INSTALACIÓN DE MIROYA-DYNACOF | 17 |
| INSTALACIÓN DE CORMAS Y MiROYA | 17 |
| INSTALACIÓN DE CORMAS | 17 |
| INSTALACIÓN DE MiROYA | 18 |
| ABRIR EL MODELO MiROYA | 18 |
| PROBAR MiROYA CON CORMAS | 19 |
| Configurar un estado inicial | 19 |
| Simular 3 años | 20 |
| Ver los indicadores | 20 |
| INSTALACIÓN DE R Y DYNACOF | 22 |
| INSTALAR EL SISTEMA R | 22 |
| INSTALACIÓN DEL SOFTWARE RSTUDIO | 22 |
| INSTALACIÓN DEL PAQUETE DYNACOF | 23 |
| 1º: instalar devTools | 23 |
| 2º : instalar Rtools | 24 |
| 2º : instalar Future | 24 |
| Descargar e instalar DynaCof | 25 |
| Probar Dynacof | 26 |
| ACOPLAR LOS DOS MODELOS | 27 |
| DESCARGAR LOS ARCHIVOS PARA EL ACOPLAMIENTO | 27 |
| CONFIGURAR LA RUTA DE ACCESO A DYNACOF (SET THE PATH) | 28 |
| EJECUTAR MiROYA SIN DYNACOF | 28 |
| ABRIR CORMAS | 28 |
| CARGAR EL MODELO MiROYA | 29 |
| PROBAR MiROYA CON CORMAS | 30 |
| Configurar un estado inicial | 30 |
| Simular 3 años | 31 |
| Ver los indicadores | 31 |
| EJECUTAR DYNACOF SIN MiROYA | 32 |

| | |
|---|-----------|
| SIMULACIÓN POR DEFECTO | 32 |
| EJECUTAR UNA SIMULACIÓN SIN PODA | 34 |
| Configurar el tipo de manejo | 34 |
| Comparar dos simulaciones | 34 |
| CONFIGURAR LOS INPUTS METEOROLÓGICOS | 35 |
| EJECUTAR MiROYA-DYNACOF CON R Y CORMAS | 37 |
| ACOPLAMIENTO YA CONFIGURADO | 37 |
| Activar el acoplamiento del lado Cormas | 37 |
| Controlar el modelo del lado R | 38 |
| Seguimiento de los indicadores | 39 |
| SIMULACIÓN DE MiROYA-DYNACOF CON SUS PROPIOS DATOS | 40 |
| Convertir los archivos de meteo para DynaCof | 40 |
| Utilizar datos de Clima de Nasa | 43 |
| Recuperar los datos diarios de un punto GPS | 43 |
| Convertir el archivo para MiRoya y DynaCof | 46 |
| Configurar MiRoya para utilizar el archivo CSV | 48 |
| Hacer la simulación S0 en DynaCof | 49 |
| Hacer una simulación completa DynaCof-MiRoya | 51 |
| SIMULACIÓN INTERACTIVA | 51 |
| CONFIGURACIÓN Y EJECUCIÓN DE UN ANÁLISIS DE SENSIBILIDAD | 53 |
| CONCLUSIÓN | 55 |
| ANEXO | 56 |
| PARÁMETROS Y SALIDAS DE DYNACOF | 56 |
| DINAMICA DEL CAFETO EN MiROYA SIN DYNACOF | 58 |
| DIAGRAMA DE ACTIVIDAD DEL CAFETO | 58 |
| FENOLOGÍA DEL CAFETO | 60 |
| REGENERACIÓN DE HOJAS | 62 |
| CAÍDA DE LAS HOJAS | 62 |
| Caída de hojas viejas | 62 |
| Caída de hojas por cosecha | 62 |
| Caída de hojas por roya | 62 |
| ACCIONES DE LA FUNGICIDA | 62 |
| Fungicida Protector | 62 |
| Fungicida curativa, Triazol | 63 |
| Results : Capa= 1 traitement/an, Umbral = 10% | 64 |

Autores que han participado en el diseño del modelo

En el marco del proyecto PROCAGICA, el diseño de este modelo requirió un largo trabajo en el que participaron varias personas:

- Jacques Avelino, CIRAD - CATIE
- Edwin Treminio, CATIE
- Pierre Bommel, CIRAD - CATIE
- Grégoire Leclerc, CIRAD - CATIE
- Natacha Motisi, CIRAD - CATIE
- Isabelle Merle, CIRAD - CATIE
- Rémi Vezy, CIRAD

Este trabajo se basa también en el conocimiento de expertos referenciado en numerosos artículos científicos.

Objetivo de MiRoya

MiRoya es un modelo para simular la dinámica de la roya y sus impactos en término de incidencia, caída de hojas y producción de café. Se trata de un modelo mecanístico, que, al poner en marcha sus mecanismos, permite **comprender y evaluar** el funcionamiento de la roya del café y los parámetros que influyen en la propagación de las epidemias. El modelo busca probar la efectividad de las prácticas que los caficultores pueden implementar.

Entonces, ante todo, *MiRoya* es un modelo científico para probar hipótesis y explorar mediante simulación la influencia de diferentes parámetros en las posibles evoluciones del sistema café-roya.

Evidentemente, no se trata de una bola de cristal que pueda predecir la evolución exacta de dicho sistema, sino más bien de un soporte informático (o quizás una muleta) para ayudar a comprender cómo las relaciones entre un huésped y un parásito pueden producir trayectorias complejas. Así pues, *MiRoya* es, en primer lugar, una herramienta que ayuda a la reflexión.

En efecto, desde el momento en que dos dinámicas se interconectan y varias entidades interactúan, es muy difícil para nuestro cerebro anticipar cómo puede evolucionar el conjunto. Y las consecuencias de una pequeña modificación pueden producir efectos imprevistos. Este es el caso del sistema de la roya del café, en el que la generación y caída de hojas afecta al crecimiento y propagación de las esporas de la roya que, a su vez, afectan al estado del cafeto. Hay que añadir que este complejo sistema depende también del clima y de las prácticas de los caficultores.

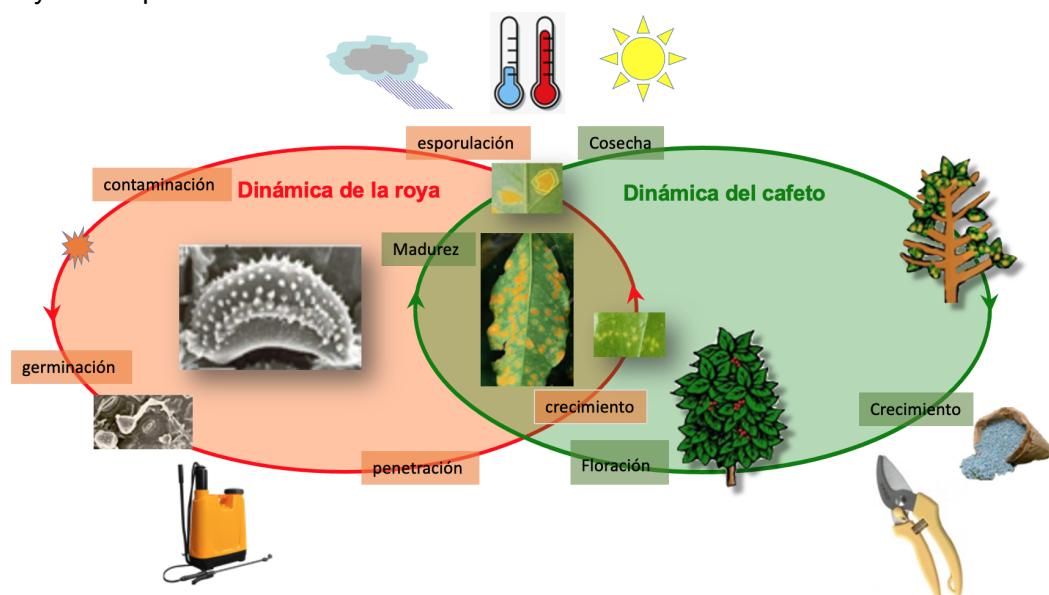


Fig 1. Diagrama de las dos dinámicas acopladas

Así, la simulación informática nos ayuda a encontrar los factores que más influyen en dicho sistema y a anticipar los posibles efectos de una acción.

Además de su dimensión puramente científica, el objetivo inicial del modelo *MiRoya* era desarrollar un **simulador interactivo** multijugador. Para apoyar la organización de juegos serios, este simulador busca promover debates sobre los aspectos socio-económicos de la producción de café en un contexto de crisis de la roya. Basado en un modelo del ciclo de vida de la roya y de los cafetos, este juego computarizado simula la producción de café según las condiciones climáticas, las enfermedades debidas a la roya y los tratamientos aplicados por los agentes productores. Dirigido a técnicos y gerentes de institutos cafeteros, el objetivo de este juego es doble: 1º) Facilitar la armonización de las alertas y de las

acciones al nivel institucional; 2º) Generar recomendaciones efectivas y oportunas para los pequeños productores con recursos financieros limitados.

Las sesiones realizadas en varios países de la región mostraron grandes disparidades en los resultados, aunque la mayoría de los participantes eran expertos en el cultivo del café y el tratamiento de la roya. Al concientiar a los participantes de las restricciones técnicas, económicas y laborales de los pequeños productores, los participantes se dieron cuenta de la imposibilidad de aplicar los fungicidas de forma sistemática. Luego se ven obligados a racionalizar sus recomendaciones y adaptarlas a las condiciones locales. Por ejemplo, tienen que elegir cuidadosamente las épocas del año en las que se debe tratar, épocas que requieren información precisa y continua sobre el clima local y el contexto socioeconómico.

Al señalar la importancia de la comunicación entre los países, este juego también tiene como objetivo estructurar una red regional de institutos meteorológicos y agronómicos. A fin de encontrar estrategias de prevención y control adaptables, también es necesario que los países intercambien información sobre los niveles de gravedad de la roya.

En conclusión, el objetivo general del modelo *MiRoya* es de integrar y probar informaciones científicas y de expertos en un modelo de simulación, para:

1. Comprender los mecanismos biofísicos
2. Jerarquizar los parámetros que influyen en las epidemias de roya
3. Compartir conocimientos,
4. Probar hipótesis,
5. Probar diferentes medios de control de la roya y
6. Consolidar una Red Regional de Gestión de Riesgos en Café (RR-GRC)

Una descripción más completa de los objetivos del modelo está disponible en este vídeo :
<https://youtu.be/D7UvZCKfv-A>

Descripción del modelo conceptual

MiRoya está basado en un modelo multi-agente (ABM en inglés) que permite simular la dinámica de la roya y sus impactos en término de incidencia, caída de hojas y producción de café. Esquemáticamente, el modelo representa fincas de café compuestas por parcelas de una hectárea. Dependiendo del clima y la época del año, crecen los cafetos y los hongos de la roya.

Estructura del modelo

El modelo conceptual se representa como en forma simple :

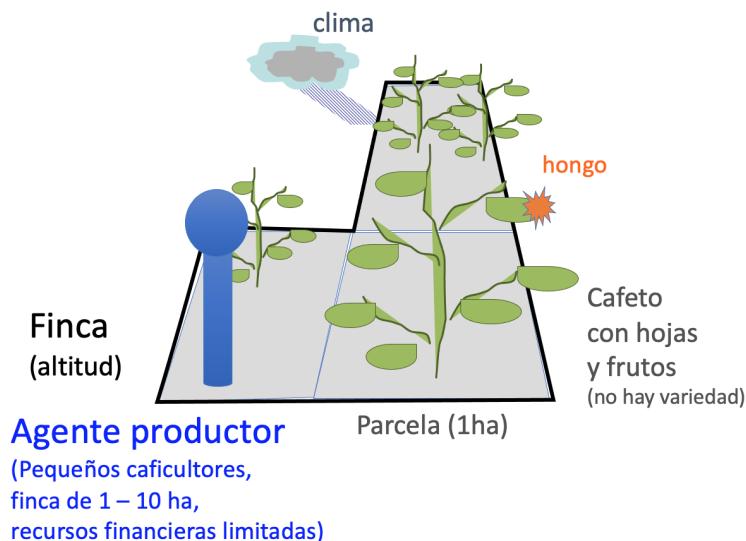


Fig. 2. Representación esquemática del modelo conceptual

Un pequeño caficultor maneja una finca de cafetos afectados por un clima local pero también por la roya. La escala espacial se limita, pues, a un pequeño territorio sometido a las mismas condiciones ambientales y climáticas. El diagrama de clase siguiente muestra la estructura del modelo (las entidades, sus atributos y relaciones):

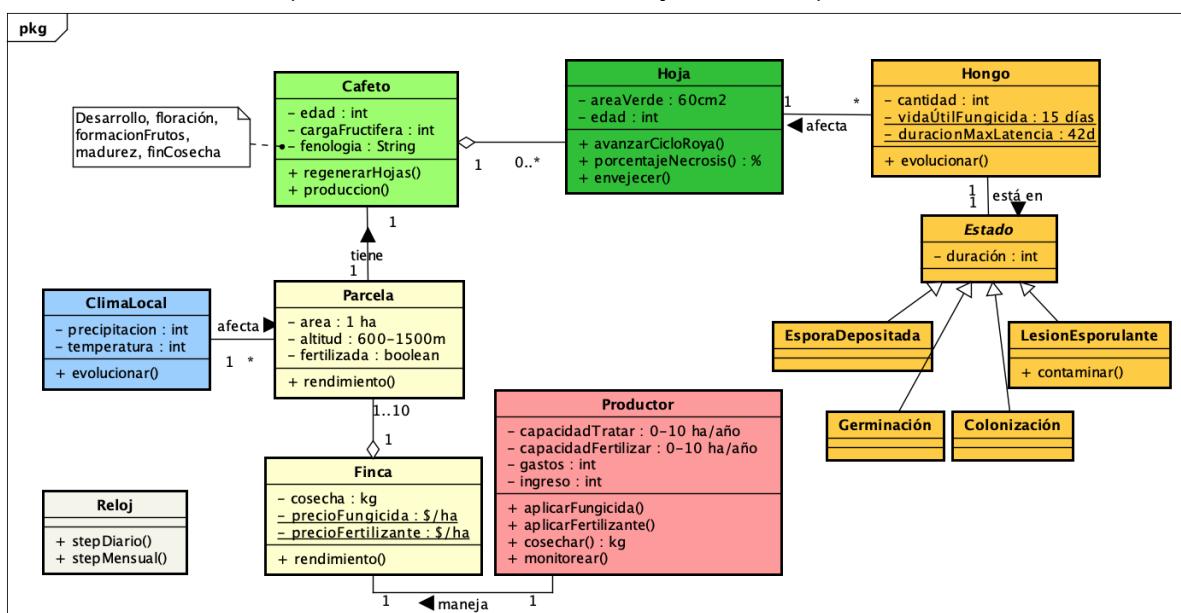


Fig. 3. Diagrama de clase UML del modelo conceptual

Un productor maneja una finca compuesta por parcelas afectadas por un clima local. En una parcela hay un cafeto, que representa un grupo de cafetos. Contiene hojas cuyo número puede fluctuar con el tiempo. En el momento de la cosecha, produce frutos.

Cada hoja es susceptible de ser infectada por la roya. Se encuentra primero en forma de esporas depositadas. Si las condiciones lo permiten, las esporas germinan y penetran en la hoja para convertirse en lesiones. Tras un periodo de latencia, las lesiones se convierten en lesiones esporulantes que emiten esporas.

Ya que en realidad hay millones de esporas liberadas en un momento dado, es imposible implementar tal modelo de manera computarizada. Los 4 posibles estados del hongo de la roya se describen, por lo tanto, mediante 4 atributos definidos a nivel de cada hoja:

1. *numDeposiciones*: un número de esporas depositadas en una hoja,
2. *numGerminaciones*: un número de esporas germinadas en una hoja,
3. *numColonizaciones*: un vector de números de lesiones no esporulantes en una hoja,
4. *numPustulas*: un vector de números de lesiones esporulantes en una hoja.

Entonces, en vez de manipular objetos (instancias de Hongos), hay 4 variables: 2 variables simples (números enteros) y 2 tablas numéricas. Las tablas (o vectores) representan el número de lesiones y sus edades (de maximum 42 o 78 días). La evolución y el envejecimiento de cada lesión se simula de la siguiente manera: cada lesión avanza de 1 hasta 3 steps cada día, en función de las condiciones climáticas. Significa que una lesión no esporulante puede estar en este estado de 14 días (en el mejor caso) hasta 42 días (en el peor caso climático). (Hongo duracionMaxLatencia = 42 días). El mismo procedimiento se aplica a las lesiones esporulantes :

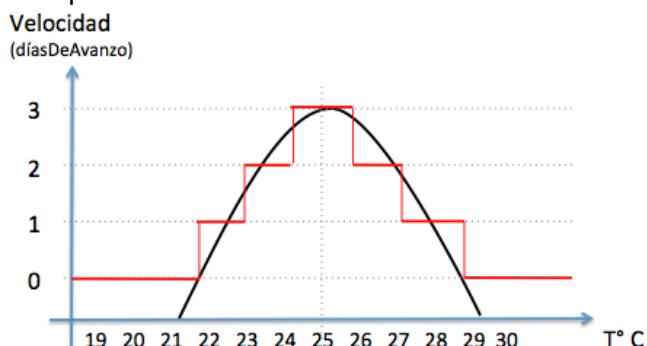


Fig. 4. Tasa de envejecimiento de una lesión en función de la temperatura

```

si 21.39233<T<23.16103 => 3 steps
si 19.62364<T<24.92972 => 2 steps
si 17.85494<T<26.69842 => 1 step, si no => 0

```

Por consiguiente, el modelo conceptual adecuado para la aplicación es el siguiente:

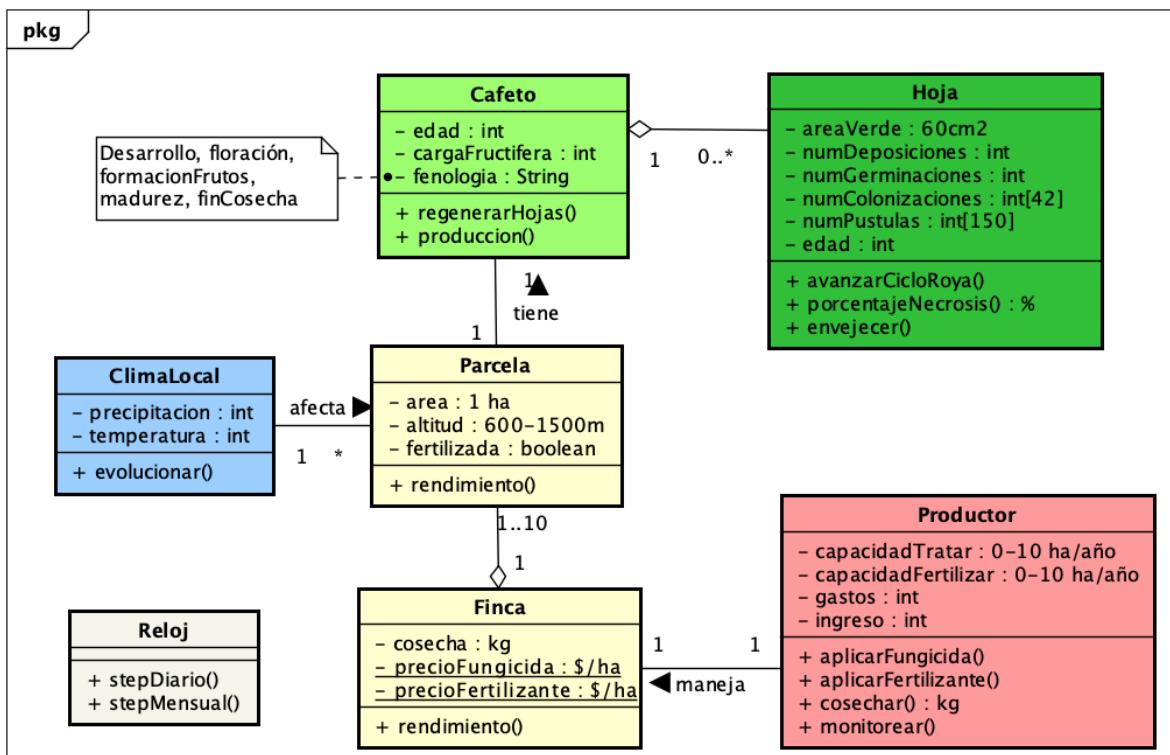


Fig. 5. Diagrama de clase UML del modelo implementado

Dinámicas del modelo

Tiempo

En la versión actual, el paso de tiempo (step) es igual a **una semana** (`stepSemanal`). Pero de hecho, la granularidad mínima es el día (`stepDiario`). Por lo tanto, el paso semanal ejecuta 7 veces un paso diario.

Este diagrama de secuencia representa la activación de la roya diariamente:

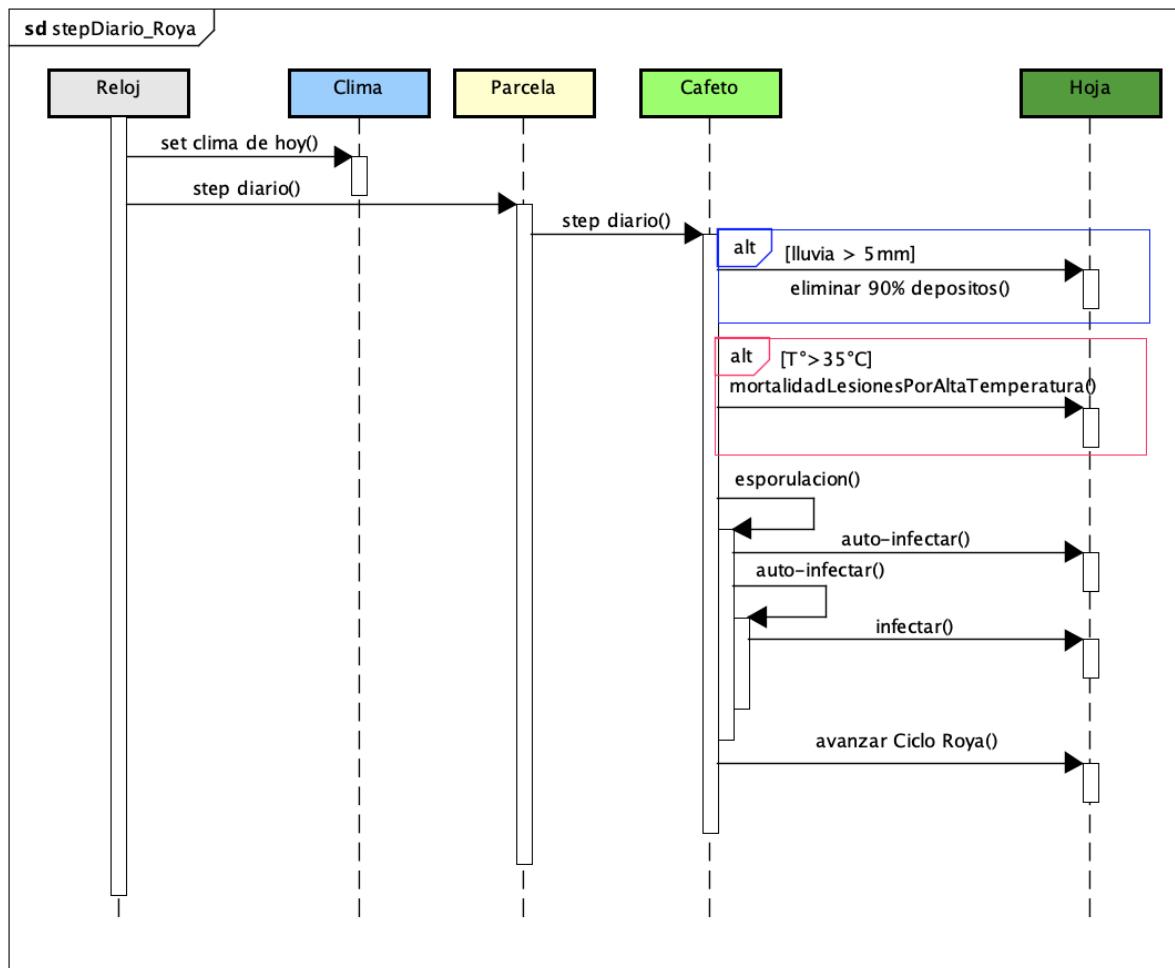


Fig. 6: Diagrama de secuencia UML del modelo (step diario) para la activación de la roya.

Este diagrama muestra que cada día la lluvia puede lavar las esporas depositadas en las hojas (si la precipitación es superior a 5mm), luego las altas temperaturas ($> 35^{\circ}\text{C}$) eliminan algunas de las lesiones, las lesiones esporulantes infectan las hojas donde se encuentran (auto-infección) y las otras hojas del cafeto, y finalmente se ejecuta el ciclo de vida de la roya (para avanzar en las etapas de las esporas).

Ciclo de vida de la roya

El siguiente diagrama de estado-transición (UML) representa el ciclo de vida de la Roya. Los 4 estados son: Espora depositada (en una hoja), Espora Germinada (con su apresorium), Lesión no Esporulante (llamada “Colonización”) y Lesión Esporulante (llamada “Pustula”). Estos dos últimos estados se encuentran dentro de la hoja.

El cambio de estado (la transición) se activa por un evento. En la mayoría de los casos, estas transiciones son activadas por eventos climáticos.

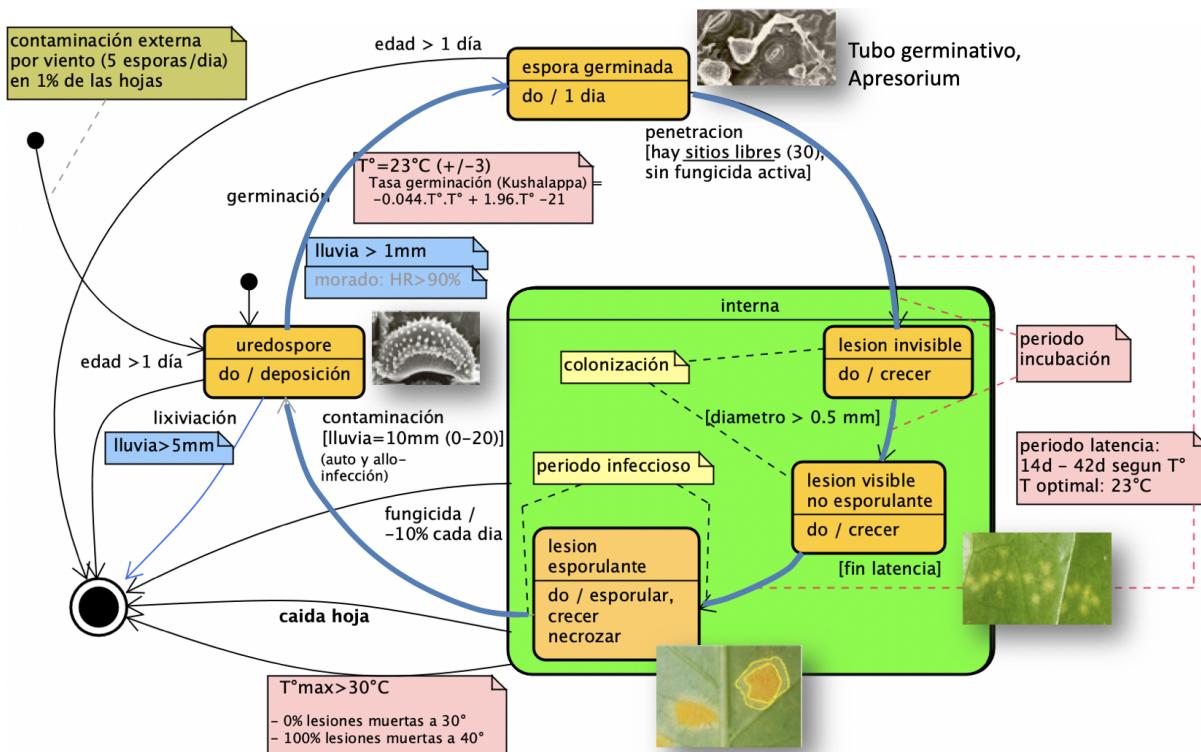


Fig. 7: Diagrama de estado-transición UML del ciclo de la roya

Como se explicó anteriormente, no hay instancias de Hongos, pero hay 4 variables: *numDeposiciones*, *numGerminaciones*, *numColonizaciones* y *numPustulas*. La traducción del diagrama de estado-transición en implementación de la roya se explica de acuerdo con el siguiente esquema:

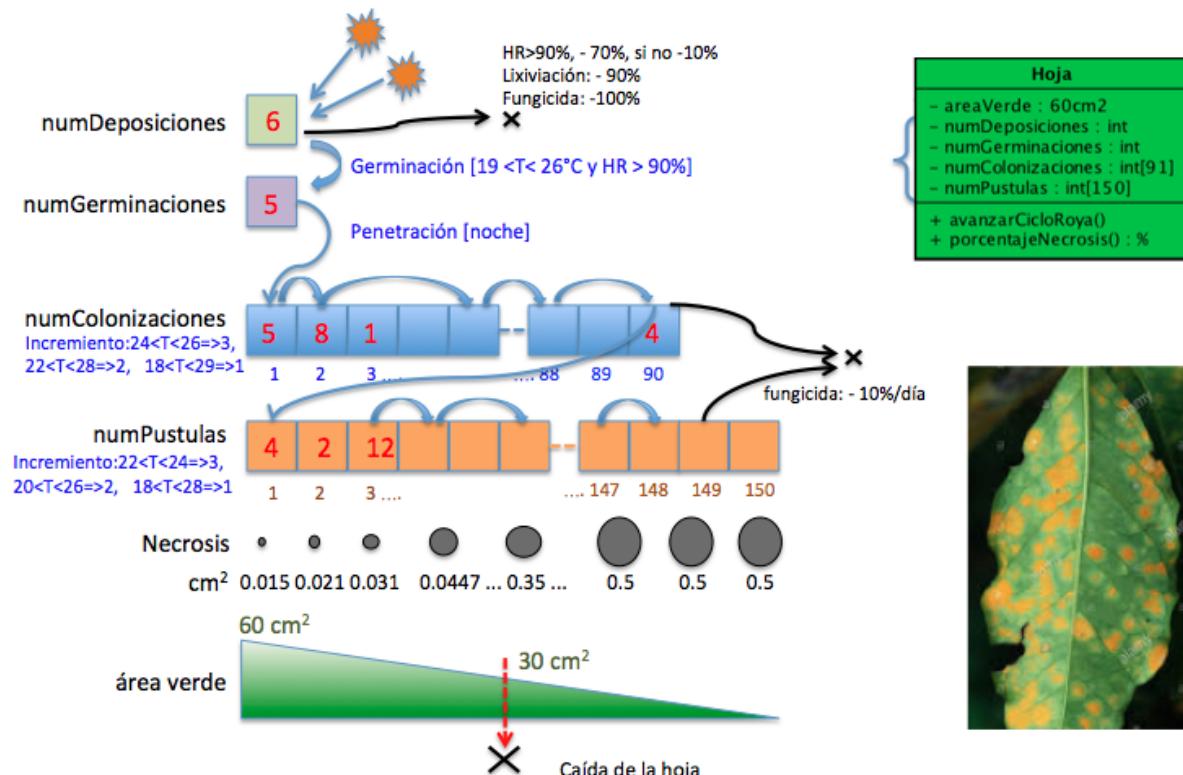


Fig. 8: La evolución de la roya sobre y dentro de la hoja

Todas las esporas que se depositan en una hoja son representadas por un número: *numDeposiciones*. Cuando las condiciones climáticas lo permiten, una parte de estas esporas van a germinar. En este caso, el valor de *numDeposiciones* se reduce a favor de la variable *numGerminaciones*. Si las condiciones lo permiten, el contenido de la variable *numGerminaciones* se transfiere y se acumula a la primera célula del vector *numColonizaciones*. En función del clima, se calcula un índice de crecimiento de las colonias. Este índice (llamado *diasDeAvanzo*) permite avanzar cada elemento del vector de 1 a 3 casillas dependiendo del clima: 3 *diasDeAvanzo* cuando la temperatura está cerca de un óptimo (23°C); 2, 1 o 0 cuanto más lejos del óptimo (cf. Fig. 4).

En siguiente, la lista de los eventos presentados en el ciclo de vida (Fig. 7) se detallan.

Lixiviación

En la versión actual, se considera solamente sistema sin sombra.

Cuando hay mucha lluvia (parámetro Cafeto umbralLixiviacion = 5 mm/dia), se elimina 90% de las esporas depositadas en todas las hojas (parámetro Hongo perdidasPorLixiviacion = 0.9).

Contaminación

Las esporas depositadas en una hoja provienen de 3 fuentes de infección: la auto-infección por la misma hoja, la alo-infección del mismo árbol y la alo-infección por viento.

Auto-infección por la misma hoja

Una hoja con lesiones esporulantes se auto-infecta. Se considera 58000 esporas emitidas por cm² de lesión esporulante (Merle et al. 2019). La mitad de la superficie de una lesión es efectivamente esporulante. Por otro lado, el 35% de las esporas liberadas son viables. Además, el 1% se deposita efectivamente en el envés de la hoja (las esporas germinadas penetran a través de los estomas de la hoja, que sólo están en la parte interna). Entonces, 1 cm² de lesión esporulante puede depositar hasta 200 esporas :

$$\text{número de esporas depositadas / cm}^2 \text{ de lesión} = 58000 * 0.35 * 0.01 = 200.$$

Además, los depósitos también dependen de la lluvia. Si la precipitación del día es mayor a 20 mm, las esporas son eliminadas. Entre 10 y 20 mm de precipitación, la proporción de depósito es óptima a 10 mm de la precipitación y es cero a 0 y 20 mm. Entre estos valores se aplica una relación lineal.

Teniendo en cuenta estos conceptos, el número de esporas depositadas y susceptibles de infectar la hoja se calcula de forma estocástica como sigue:

$$\begin{aligned} \text{número de esporas depositadas} &= \text{random entre: 1 y:} \\ &(\text{proporción_por_luvia} \times 200 \times \text{area de pustulas}). \end{aligned}$$

Alo-infección por las hojas vecinas del mismo árbol

Para la propagación de las esporas de una hoja a otra del árbol, se utilizó una ley de probabilidad de La Place. Bullock et al., (2006)¹ explican que “El Laplace resulta de los propágulos que se mueven aleatoriamente y tienen una cierta probabilidad de asentamiento por unidad de tiempo. [...] Estas funciones de densidad de probabilidad describen la densidad de los propágulos a una distancia x de una fuente” :

¹ Bullock, J. M., Shea, K., & Skarpaas, O. (2006). Measuring plant dispersal: an introduction to field methods and experimental design. *Plant Ecology*, 186(2), 217-234.

$$P(x) = \text{coef_LaPlace} \cdot 1/2b \cdot \exp(-|x-a|/b)$$

con $b = 2$, $a = 0$, $\text{coef_LaPlace} = 0.125$, obtenemos las siguientes probabilidades:

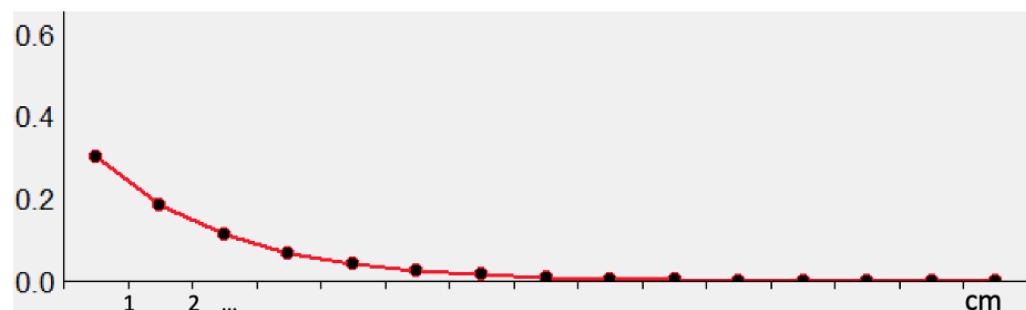


Fig. 9: Ley de probabilidad de La Place

Esta probabilidad también se multiplica por la proporción debida a la lluvia, misma proporción del cálculo utilizado anteriormente, con un depósito óptimo a 10 mm de la precipitación.

Alo-infección externa por viento

Además de estos dos procesos, cada día, hay 10% de chance (`Hongo probalInfeccionViento = 10%`) que 1% (`Hongo proporcionHojasViento`) de las hojas (seleccionadas al azar) reciben 5 esporas del exterior (`Hongo numDeposicionPorHoja = 5`). Esto representa el ruido de fondo difuso de las esporas en el aire.

Germinación

En el laboratorio, la viabilidad de las esporas es muy baja después de 15 días de almacenamiento. En condiciones naturales, este tiempo se reduce aún más. Una vez depositada, la espora tiene muy poco tiempo para germinar y penetrar. Si las condiciones son húmedas, lo hace en 48 horas, siempre que haya agua abierta. Si las condiciones no son buenas para la germinación y la penetración (es decir, menos de 6 horas de humedad), la espora comienza a germinar pero no completa el proceso y se muere.

Según Kushalappa, el porcentaje de germinación es una función cuadrática de la temperatura (T):

$$\% \text{ germinacion} = a \cdot T^2 + b \cdot T + c = -0.044087 \cdot T^2 + 1.964224 \cdot T - 21.016217$$

Para tener en cuenta la humedad, consideramos que esta ecuación sólo es aplicable si la precipitación es superior a 1 mm.

Penetración

La penetración dura de 1 a 5 horas en la noche, pero no todas las esporas germinadas pueden entrar. Sólo se dispone de 30 sitios de infección por hoja. Entonces la integración de nuevas lesiones en la hoja se limita al número de sitios disponibles, sabiendo que cada lesión (no esporulante y esporulante) ocupa 1 sitio.

Período de latencia y de esporulación

Como se explicó anteriormente, en función del clima, se calcula un índice de crecimiento de las colonias. Este índice (llamado `diasDeAvanzo`) permite avanzar cada elemento del vector de 1 a 3 casillas dependiendo del clima: 3 `diasDeAvanzo` cuando la temperatura está cerca de un óptimo (23°C); 2, 1 o 0 cuanto más lejos del óptimo (cf. Fig. 4).

Esporulación

Para salir del estado “no esporulante” al estado “esporulante” (o pústula), las lesiones tienen que esperar entre 14 hasta 42 días de acuerdo con las condiciones climáticas. Durante este período, ellas no afectan la hoja (no es el caso en realidad).

A 23°C (temperatura ideal), cada lesión salta 3 días de avance. Con esta temperatura, necesita de 14 días para entrar en el estado pústula. Cuando la temperatura es menor de 18°C o mayor de 28°C, necesita de 42 días para convertirse en pústula.

Una hoja que contenga una o más lesiones esporulantes puede autoinfectarse e infectar otras hojas del cafeto (como se describe en el párrafo anterior “Contaminación”).

Duración de la lesión esporulante

El proceso de avanzar en los días de lesión esporulante es parecido a lo de lesión no esporulante, excepto que la duración puede ser más grande (hasta 150 días). Entonces, cuando la temperatura es menor de 18° o mayor de 28°, una lesión esporulante puede vivir hasta 150 días.

Mortalidad por alta temperatura

El exceso de temperatura mata las lesiones. El modelo considera que una temperatura máxima del aire por encima de los 35°C (Hongo tempMinMortalidad) es excesiva. A 40°C durante 4 días, todas las lesiones de una hoja mueren. Entre estas dos temperaturas, la tasa de mortalidad es una función lineal de la temperatura (mortalidad = 0% a 35° - 100% a 40°C) durante 4 días (Ribeiro et al., 1978)².

Como el step es de un día (y no de 4 días), el modelo utiliza la siguiente función (se utiliza la T° máxima del aire):

Si $tMax \geq 40^\circ$, eliminar el 75% de las lesiones (Hongo propMortalidadTmax = 40°C).

Si $35 < tMax < 40^\circ$, entonces eliminar una proporción de las lesiones.

Dinamica del Cafeto

En su versión anterior, el modelo MiRoya proponía un mecanismo simplista de crecimiento de las hojas (ver el anexo “[Dinamica del Cafeto en MiRoya sin DynaCof](#)”). Pero los análisis mostraron que la dinámica del huésped juega un papel muy importante en el sistema huésped-parásito.

Por esta razón se decidió acoplar MiRoya con un modelo de crecimiento del cafeto de probada eficacia: el modelo DynaCof (Vezy et al., 2019)³.

DynACof calcula la Productividad Primaria Neta a escala de parcela, la asignación de Carbono, el crecimiento, el rendimiento, la energía y el balance hídrico del cafeto. El modelo también utiliza cohortes de flores y de frutas para el desarrollo reproductivo a fin de representar mejor la distribución de la demanda de C de las frutas a lo largo del año.

DynACof está basado en la ley del mínimo entre la oferta y la demanda de C, en el siguiente orden:

1. La madera de crecimiento, las ramas y las raíces grandes se alimentan primero.
2. Los botones florales están alimentados. Se iniciaron durante el año anterior (adaptado del modelo Rodriguez et al., 2011), en función de la relación entre los nodos y el **LAI (Leaves Area Index = índice de área foliar)** y la T°.

² Ribeiro et al. (1978), Efeito de alta temperatura no desenvolvimento de Hemileia Vastatrix em cafee suscetivel. Bragantia, Revista Cientifica do Instituto Agronomico do Estado de São Paulo

³ Vezy R., le Maire G., Christina M., Georgiou S., Imbach P., Hidalgo H, Alfaro E., Blitz-Frayret C., Charbonnier F. Lehner P., Loustau D. & Roupsard O. (2019). « DynACof: a process-based model to study growth, yield and ecosystem services of coffee agroforestry systems ». 10.5281/zenodo.3246268.

3. Frutos: se alimentan según un % de frutos maduros y según el número de botones florales. Los frutos provienen de flores que crecen en cohortes de acuerdo a una función logística.
4. Hojas: producen la oferta de C. (mortalidad y/o regeneración)
5. Raíces finas.
6. El resto de C recarga las reservas. C de reservas contribuye al cálculo de la oferta de C en el siguiente ciclo.

Este modelo incorpora numerosos factores que pueden agruparse en tres partes: el suelo, el café y los árboles de sombra, como se muestra en la siguiente figura :

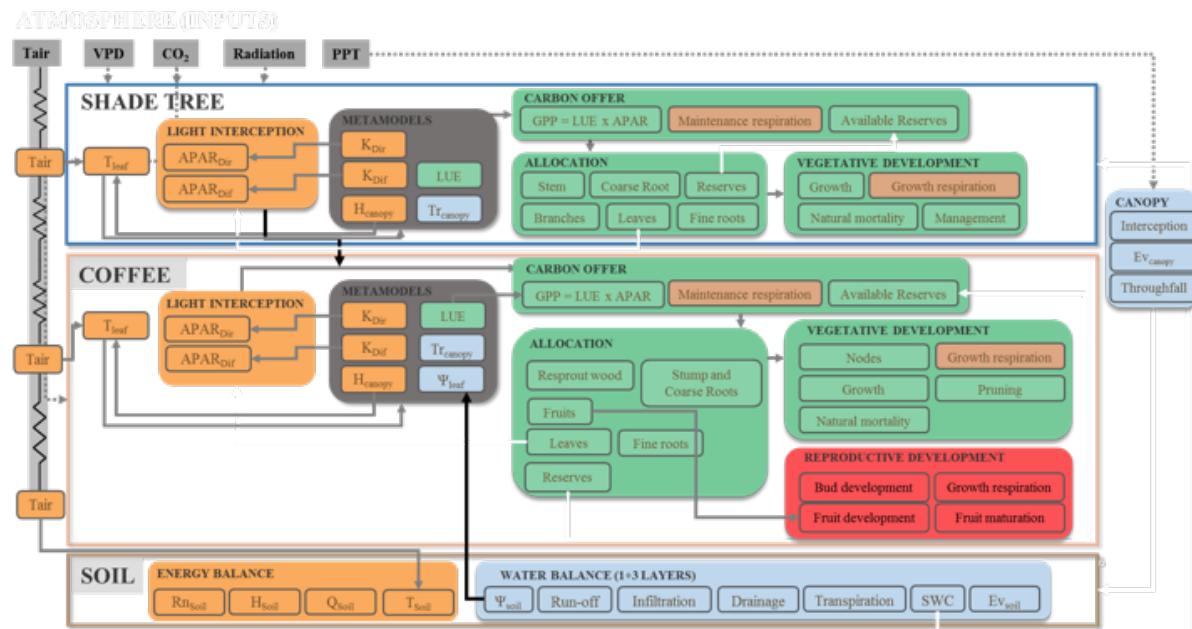


Fig. 10: Organización general de DynaCof

La lista de los parámetros y de las variables de salida de DynaCof está disponible en anexo [“Parámetros y salidas de DynaCof”](#).

Para una descripción más completa del modelo, ver Vezy et al. (2019).

Caída de la hoja por la roya (relación estadística)

A diferencia de la versión inicial, la actual utiliza una relación estadística para determinar la caída de las hojas por la roya. Esta relación estadística está tomada a partir de las observaciones realizadas por Jacques Avelino (no publicado).

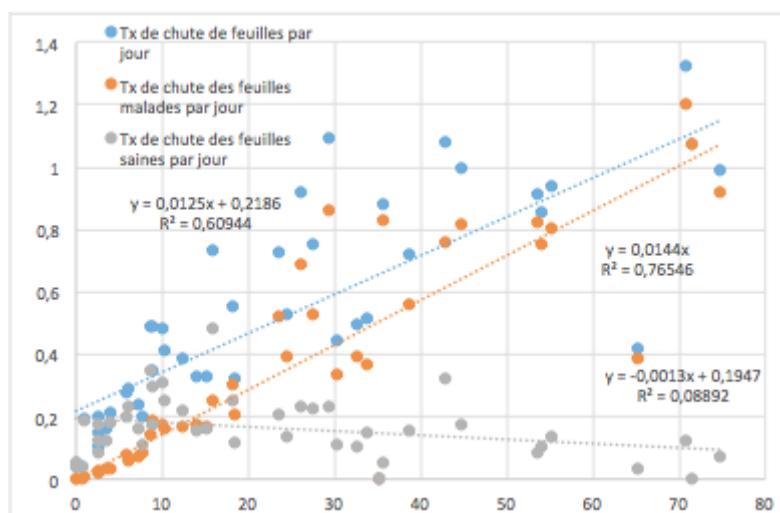


Fig. 11: Relación estadística entre la tasa de caída de las hojas y la incidencia de la roya

Este gráfico muestra el porcentaje de hojas que caen cada día en función de la incidencia de la roya (eje de abscisas). Aquí la incidencia se calcula sobre el número total de lesiones (esporulantes y no esporulantes). Si sólo se consideran las hojas infectadas, se observa una relación lineal

$$y (\% \text{ de hojas que caen/día}) = 0.0144 \times \text{incidencia}$$

Esta ecuación se utiliza para eliminar las hojas con más lesiones esporulantes (si el cafeto no tiene lesiones esporulantes, no se elimina ninguna hoja). Nota que la mortalidad natural de las hojas (así como por poda) está determinada por DynaCof.

Implementación del modelo

Como anteriormente se explicó, MiRoya-DynaCof se compone de dos submodelos: *MiRoya*, implementado en la plataforma Cormas (<http://cormas.cirad.fr> y codificado en Smalltalk), y *Dynacof*, implementado en R. *Dynacof* se encarga del crecimiento de las hojas y *MiRoya* se encarga del ciclo de vida de la roya y sus efectos en la caída de las hojas.

El paso temporal de *Dynacof* y *MiRoya* es el día. Pero como los intercambios entre los dos modelos son lentos, las interacciones ocurren cada semana. Como se describe en la figura 11, un ciclo completo de una semana consta de 6 pasos:

1. R roda 7 días de *DynACof* qui calcula (entre otras cosas) el crecimiento foliar,
2. R llama Cormas con nuevo valor de LAI (índice de área foliar)
3. *MiRoya* adapta el número de hojas según LAI
4. Cormas roda 7 días de *MiRoya*: ciclo de vida de los hongos,
5. *MiRoya* determina las pérdidas de hojas por roya
6. Cormas retorna un valor de LAI a *DynACof*. La masa de carbono (CM_leaf = leaf C mass, en g/m²/d) se modifica calculando la diferencia de LAI en el tiempo *t* y en el tiempo *t* - 7:

$$\text{CM_leaf}(t) \leftarrow \text{CM_leaf}(t) . [\text{LAI}(t) - \text{LAI}(t-7) / \text{LAI}(t-7)]$$

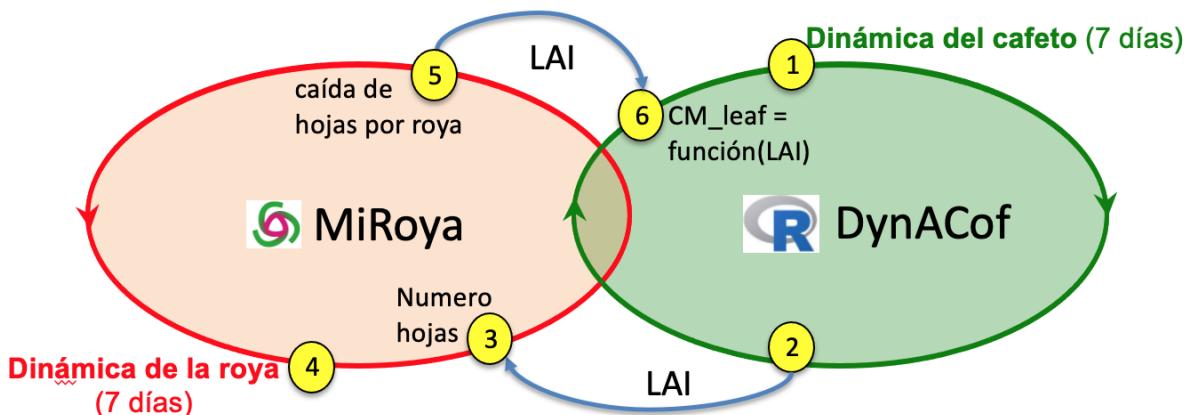


Fig. 12: Acoplamiento de las dinámicas entre DynaCof y MiRoya

De manera más formal, el siguiente diagrama de actividades UML describe la organización de un step semanal completo:

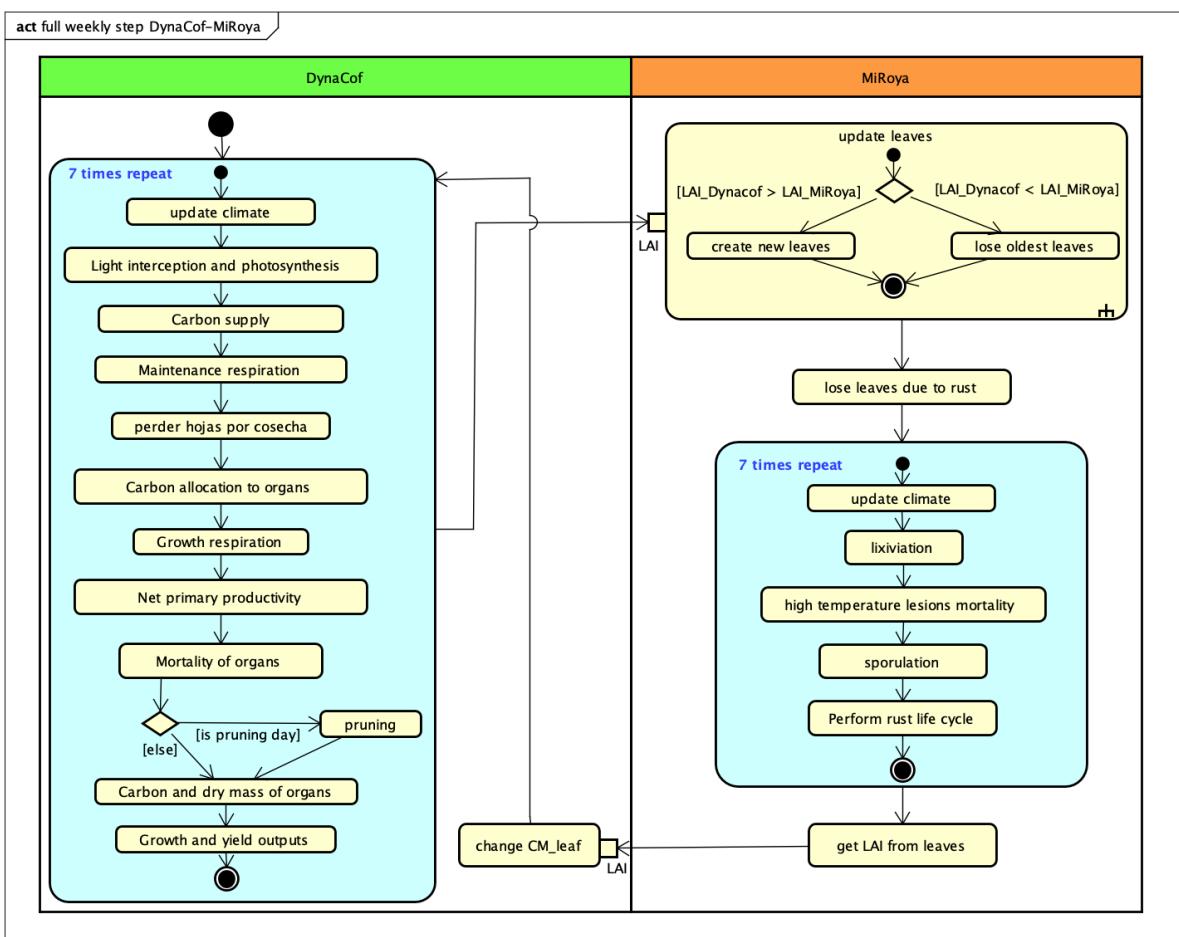


Fig. 13: Diagrama de actividades de un step (1 semana) entre DynaCof y MiRoya

Instalación de MiRoya-DynaCof

Instalación de Cormas y MiRoya

El modelo *MiRoya* que simula la incidencia de la roya es ejecutable en la plataforma Cormas. (<http://cormas.cirad.fr>).

Instalación de Cormas

Cormas es un software libre y de código abierto desarrollado por el CIRAD. Se puede descargar en el siguiente sitio: <http://cormas.cirad.fr>

Para una instalación completa de Cormas, siga las instrucciones de la página: <http://cormas.cirad.fr/enoutil/download/>

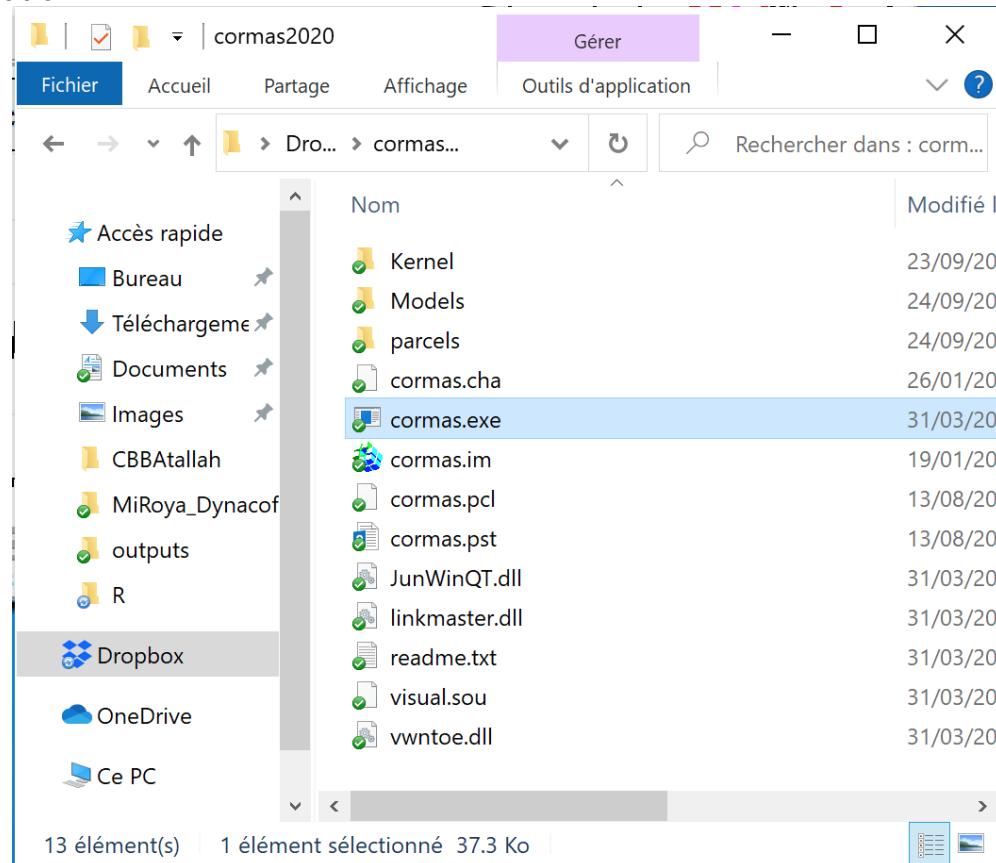
Si no, el siguiente archivo es para una instalación mínima de VisualWorks y Cormas.

Si no, descargue [cormas2020_package.zip](#) (22 Mo) y descomprimalo en la carpeta que deseé. Una copia de Cormas también está disponible en la página de Pergamino (archivo cormas2020_package.zip).



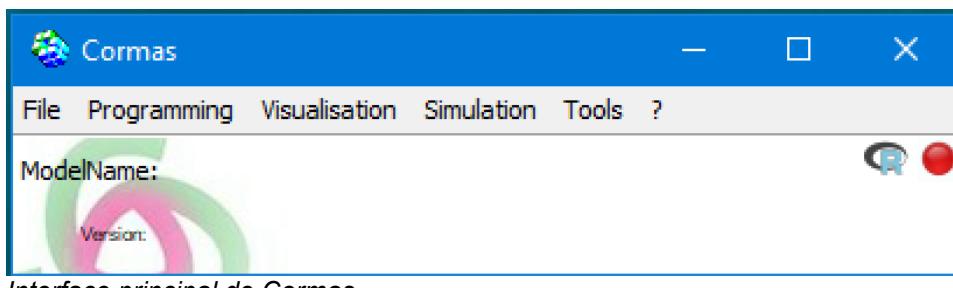
http://cormas.cirad.fr/logiciel/cormas2020_package.zip

Si se ha descomprimido en la carpeta D:/Dropbox/ (por ejemplo), se obtiene la siguiente organización:



A continuación, ejecute **cormas.exe**.

Nota: esta instalación contiene cormas.exe para Windows. También funciona en Mac y Linux, pero en este caso es necesario abrir el archivo cormas.im asociándolo a cormas/macx/visual.app



Instalación de MiRoya

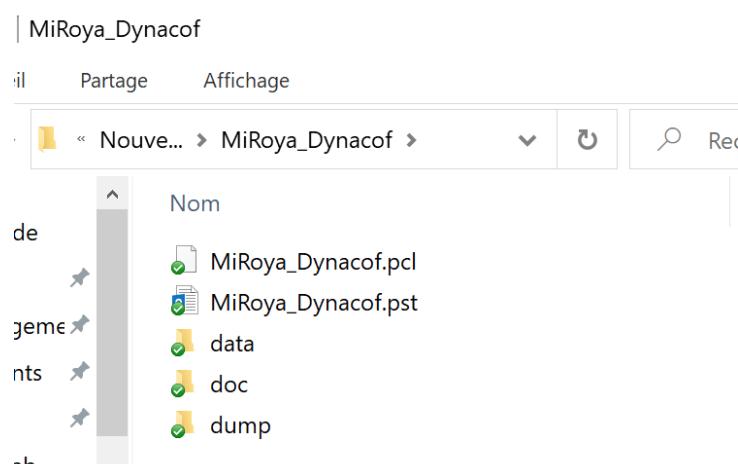
El código de MiRoya está disponible aquí :

https://github.com/pergamino/project/raw/master/Modulo-MiRoya/MiRoya_Dynacof.zip

Debe descomprimir este archivo en la carpeta **Cormas2020/Models/**.

› cormas2020 › Models › MiRoya_Dynacof ›

La estructura de la carpeta MiRoya_Dynacof es la siguiente:

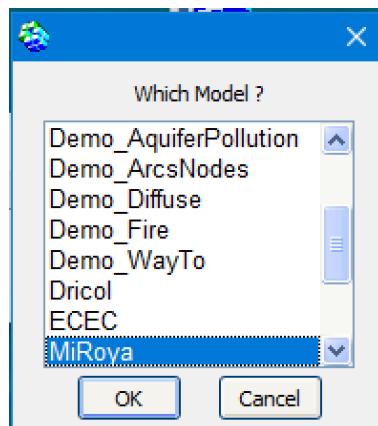


La carpeta 'data' contiene los archivos meteorológicos y epidémicos que utilizará el modelo.

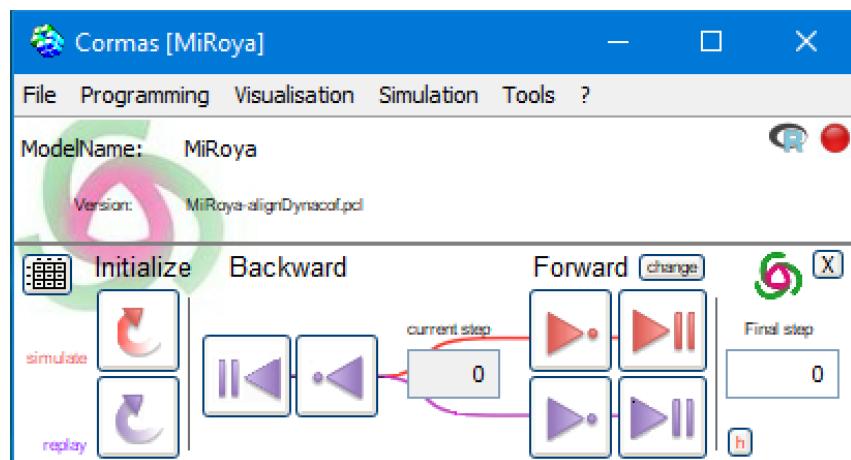
Abrir el modelo MiRoya

Se trata de probar una versión sencilla del modelo MiRoya sin acoplamiento con DynaCof.

Desde la interfaz principal de Cormas, ir al menú "File" y luego "Load":



Luego seleccione el archivo **MiRoyapcl** y haga clic en OK. El modelo se carga así en Cormas :



Probar MiRoya con Cormas

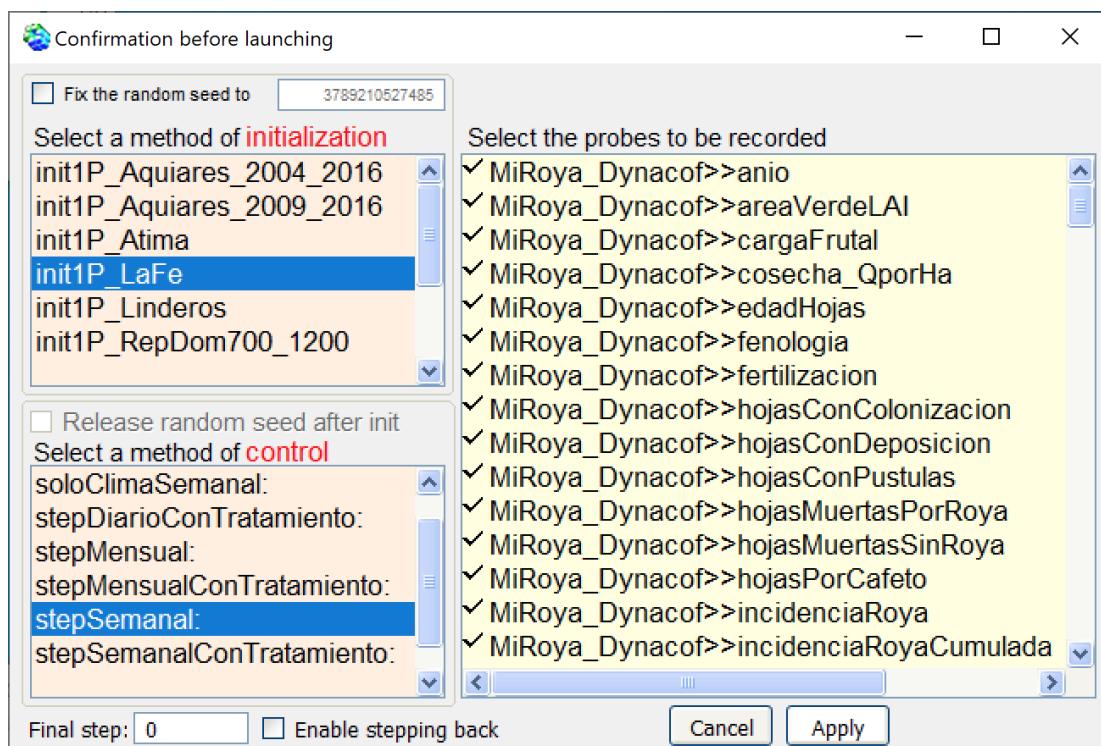
Se trata de probar una versión simplificada del modelo, sin el acoplamiento con DynaCof. El objetivo es sólo comprobar que el modelo se puede ejecutar en Cormas. El capítulo « Ejecutar MiRoya sin DynaCof » (p. 28) detalla cómo utilizar el modelo, con o sin acoplamiento.

Configurar un estado inicial

Para simular MiRoya, primero debes crear un primer estado del "mundo". Esto se hace haciendo clic en el botón rojo llamado "**simulate**" :



Se abre una nueva interfaz de inicialización:

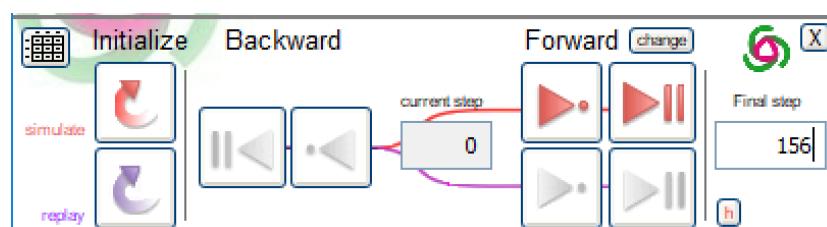


Interfaz para la construcción de un escenario de simulación

En esta interfaz se debe seleccionar el método de inicialización (por ejemplo, *init1P_LaFe*) y un método de control (por ejemplo, *stepSemanal*). También tiene que seleccionar los indicadores que queremos seguir durante una simulación. Por lo tanto, deben ser seleccionados en el lado derecho de la interfaz (zona amarilla). Para seleccionarlos todos, haz clic en el primero y luego en el último con la tecla shift pulsada. Luego haga clic en "Apply".

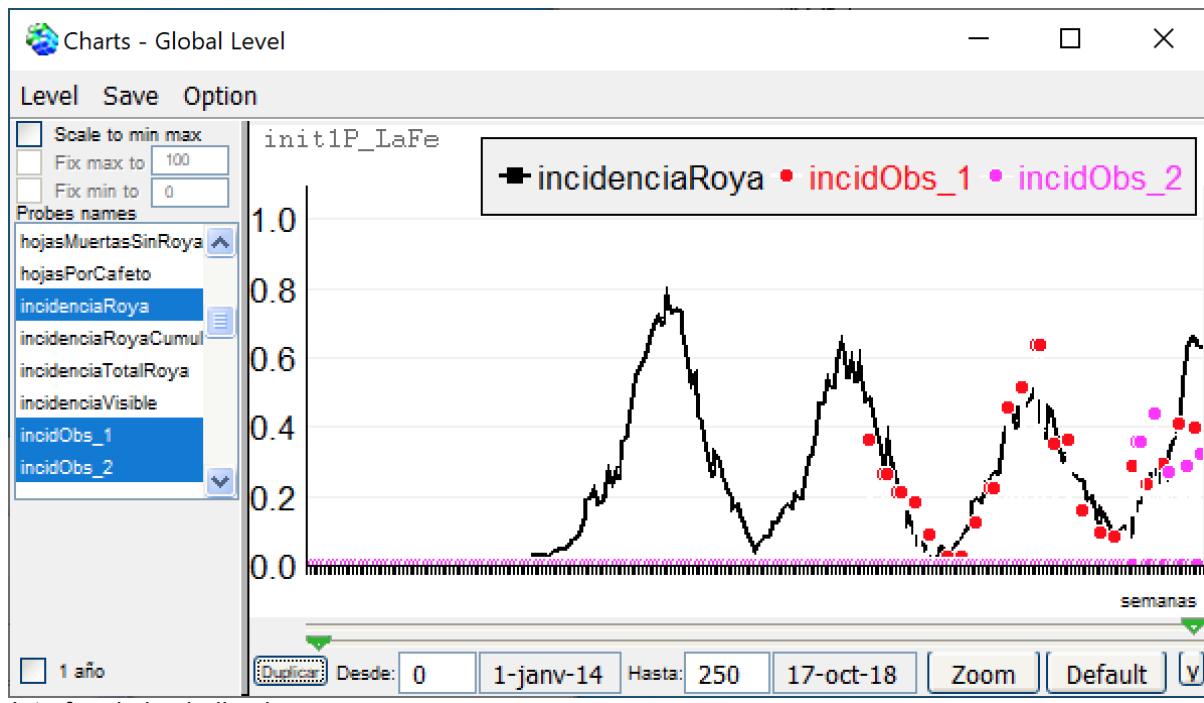
Simular 3 años

Cada clic en el botón rojo "Forward" hace que se ejecute una semana de simulación. Si queremos simular 3 años, ¡tendríamos que hacer clic 156 veces (3x 52)! En cambio, para correr 156 semanas seguidas, es mejor introducir el valor final (156 en el "Final step") y luego hacer clic en el botón rojo de avance rápido .



Ver los indicadores

Para ver los indicadores (también llamados sondas o probes en inglés), ir al menú "Visualisation" -> "Probes". A continuación, seleccione los indicadores de su elección:



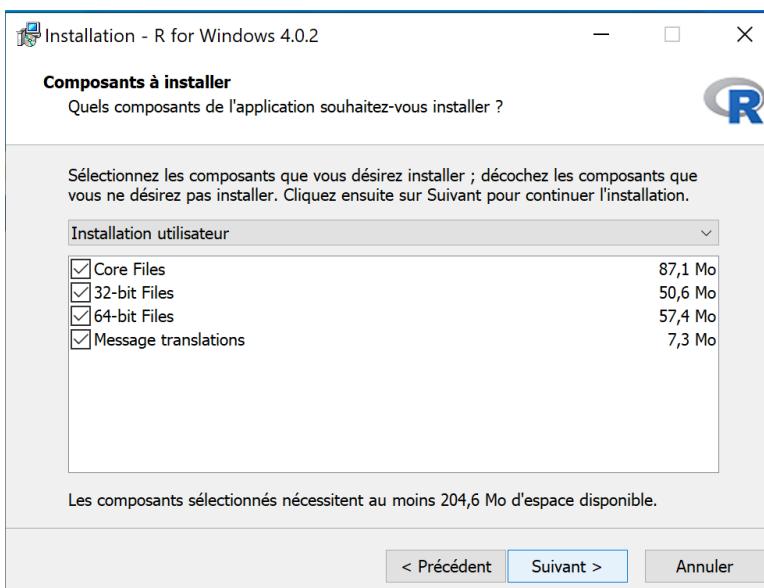
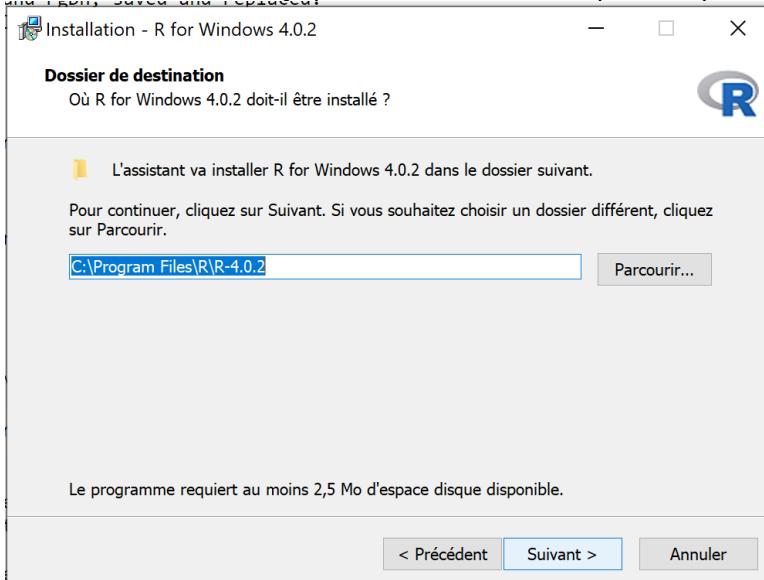
Instalación de R y DynaCof

DynaCof es un modelo implementado en el lenguaje R. Para utilizar DynaCof, es por tanto necesario que R esté instalado en su computadora.

Instalar el sistema R

Para una instalación en Windows, es necesario ir a esta página: <http://cran.r-project.org/bin/windows/base/>

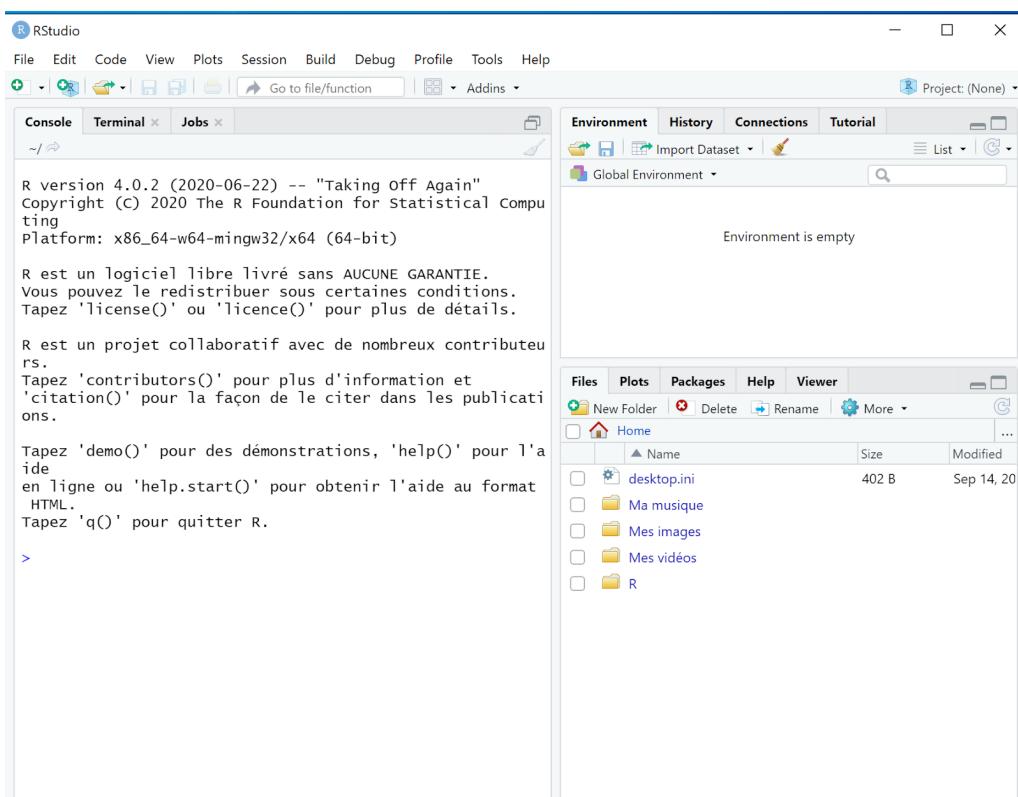
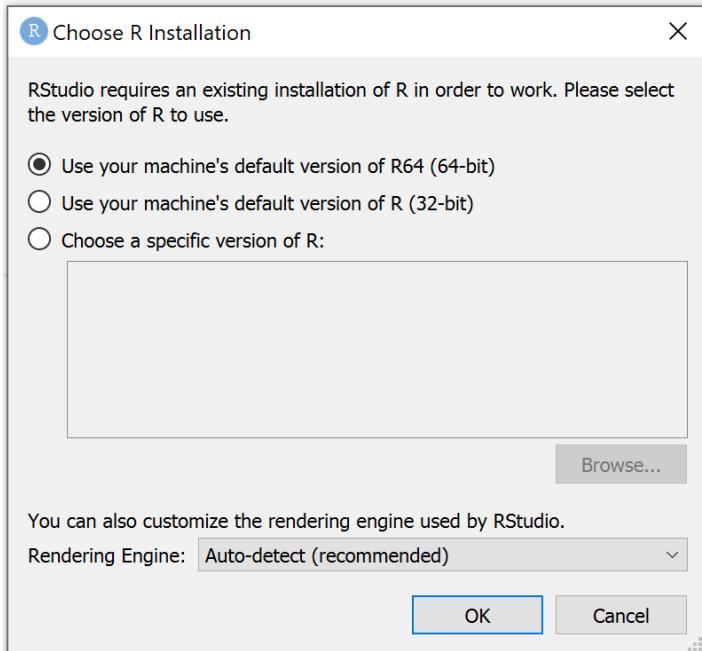
A continuación, siga el primer enlace para descargar el instalador. Una vez lanzado el programa de instalación, sólo hay que instalar R con las opciones por defecto.



Instalación del software RStudio

Una vez que R esté bien instalado, accede a <http://www.rstudio.com/products/rstudio/download/> vaya aquí para descargar la última versión de RStudio. En concreto, se trata de la edición de código abierto de RStudio Desktop (de hecho, también existe una versión de servidor).

Eige el instalador para su sistema operativo y sigue las instrucciones del instalador.

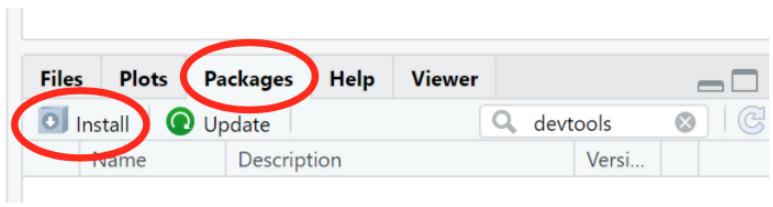


Instalación del paquete DynaCof

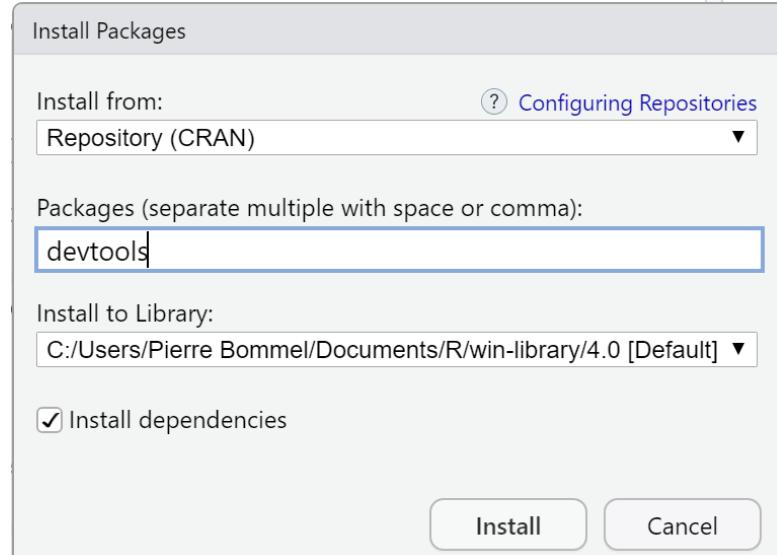
Para instalar el modelo *DynaCof*, primero es necesario instalar herramientas R: devTools, RTools y Future.

1º: instalar devTools

Primero, haga clic en la pestaña "Packages" en la esquina inferior derecha de la ventana de RStudio:



y hacer clic en el botón "Install", luego escribir "devtools" y pulsar el botón "Install":



El paquete solicitado se busca en el sitio CRAN y se descarga en su máquina. Esto puede llevar tiempo...

```
package 'testthat' successfully unpacked and MD5 sums checked
package 'withr' successfully unpacked and MD5 sums checked
package 'devtools' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\Pierre Bommel\AppData\Local\Temp\RtmppekKDDm\downloaded_packages
> |
```

2º : instalar Rtools

Repetir el procedimiento anterior para el paquete RTools: “Rtools is required to build R packages” : <https://cran.rstudio.com/bin/windows/Rtools/>

Starting with R 4.0.0 (released April 2020), R for Windows uses a brand new toolchain bundle called **rtools40**.

This version of Rtools upgrades the mingw-w64 gcc toolchains to version 8.3.0, and introduces a new build system based on [msys2](#), which makes easier to build and maintain R itself as well as the system libraries needed by R packages on Windows. For more information about the latter, follow the links at the bottom of this document.

This documentation is about rtools40, the current version used for R 4.0.0 and newer. For information about previous versions of Rtools that can be used with R 3.6.3 or older, please visit [this page](#).

Installing Rtools40

Note that rtools40 is only needed to build R packages with C/C++/Fortran code from source. By default, R for Windows installs the precompiled “binary packages” from CRAN, for which you do not need rtools!

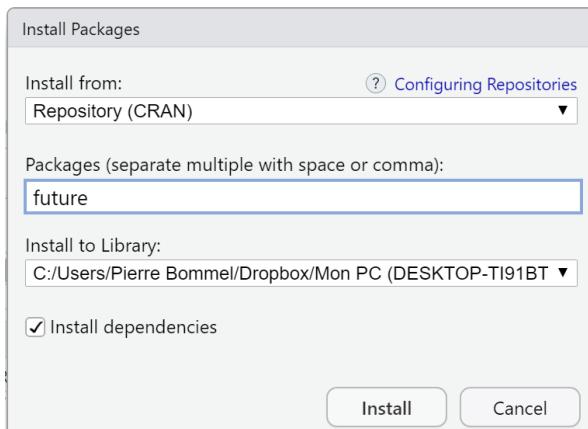
To use rtools40, download the installer from CRAN:

- On Windows 64-bit: [rtools40-x86_64.exe](#) (recommended: includes both i386 and x64 compilers)
- On Windows 32-bit: [rtools40-i686.exe](#) (i386 compilers only)

```
writeLines('PATH="${RTOOLS40_HOME}\usr\bin;$PATH"', con = "~/.Renviron")
```

2º : instalar Future

Repetir el procedimiento anterior para el paquete “Future” :



Descargar e instalar DynaCof

El modelo *DynaCof* está disponible en GitHub en la siguiente página:
<https://github.com/VEZY/DynACof>

Nota: si ya está instalada una versión antigua de DynaCof, deberá eliminarla primero desde la consola R:

```
detach("package:DynACof", unload = TRUE)
```

Hay 3 formas posibles de instalar DynaCof desde R :

```
remotes::install_github("VEZY/DynACof")
```

o:

```
devtools::install_github("VEZY/DynACof")
```

o, si ya ha descargado el package DynaCof en su máquina :

```
install.packages("C:/Users/bommel/cormas2019/R/DynACof_1.2.0.tar.gz", repos = NULL,
type="source")
```

Durante la instalación puede aparecer el siguiente mensaje:

```
package 'stringi' successfully unpacked and MD5 sums checked
Erreur : Failed to install 'DynACof' from GitHub:
(converti depuis l'avis) cannot remove prior installation of package 'stringi'
```

No se preocupe, viene de RStudio, que no gestiona bien las dependencias (ocurre a menudo). En este caso, hay que instalar los paquetes a mano. Para ello, hay que escribir, por ejemplo :

```
install.packages("stringi")
```

Al final, se instala la librería y se copia el paquete en el disco, por ejemplo:
C:\Users\bommel\Documents\R\win-library\3.6\DynACof

```

varnames          ntm!
warn.var          html
write.results    html
## building package indices
## testing if installed package can be loaded from temporary location
### arch - i386
### arch - x64
## testing if installed package can be loaded from final location
### arch - i386
### arch - x64
## testing if installed package keeps a record of temporary installation path
* DONE (DynaCof)

```

Probar Dynacof

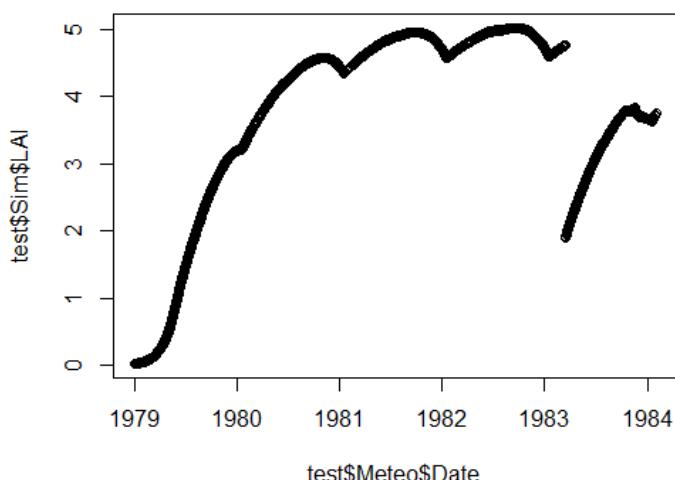
Se trata de probar una versión sencilla del modelo DynaCof sin acoplamiento con MiRoya. En una consola de R, escribir:

```

rm(list = ls())
library("DynaCof")
Sys.setenv(TZ="UTC")
sim = DynACof(Period= as.POSIXct(c("1979-01-01", "1985-12-31")))
plot(sim$Meteo$Date,sim$Sim$LAI)

```

Como resultado, se obtiene:



Muestra la evolución del LAI (Leaf Area Index = Índice de Área Foliar) durante 5 años. Nótese que se ha realizado una poda a principios de 1983.

Acoplar los dos modelos

Descargar los archivos para el acoplamiento

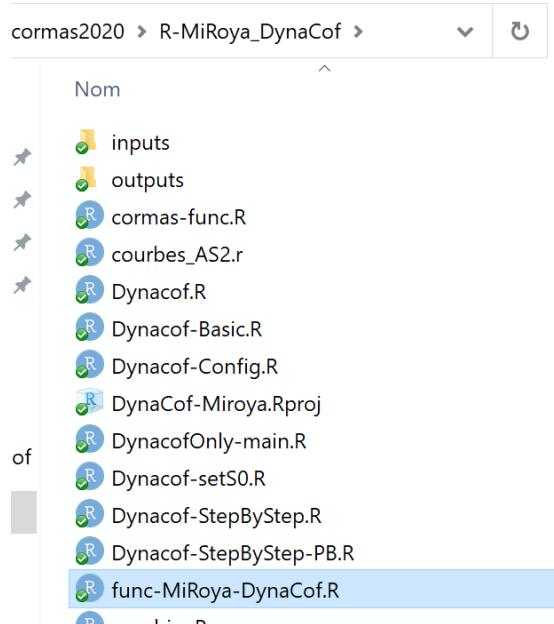
DynaCof y MiRoya pueden funcionar de forma independiente. Pero para acoplarlos, es necesario descargar el siguiente archivo zip :

https://github.com/pergamino/project/raw/master/Modulo-MiRoya/R-MiRoya_DynaCof.zip

Debe descomprimir este archivo en la carpeta **/Cormas2020/**.

» cormas2020 » R-MiRoya_DynaCof »

La estructura de la carpeta R-MiRoya_Dynacof es la siguiente:



La carpeta contiene los archivos R para ejecutar DynaCof y MiRoya juntos. También contiene las carpetas "inputs" y "outputs".

Por ejemplo, en RStudio, abrir el archivo main-MiRoya-Dyn LAI LaFe.R :

The image shows the RStudio interface with the code editor open. The title bar says 'main-MiRoya-Dyn LAI LaFe.R'. The code in the editor is as follows:

```

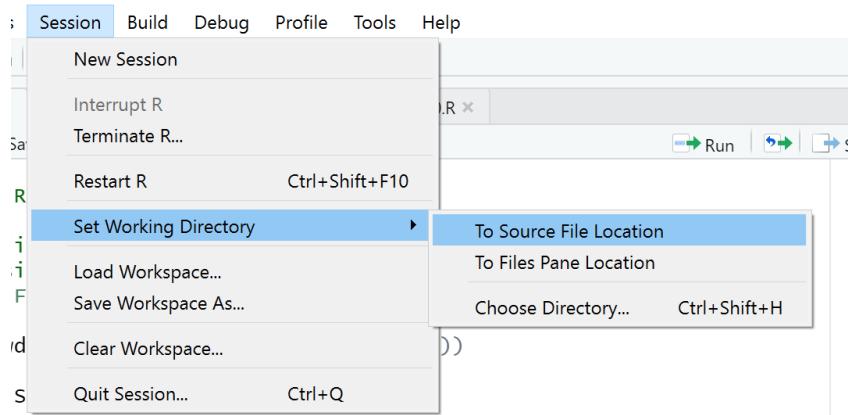
1 rm(list = ls())
2 source("func-MiRoya-DynaCof.R")
3 #parameters
4 cormasInit <- "init1P_LaFe"
5 s0_sim <- "s0_sim_LaFe_2014-2020.rda" #Name of the basic simulation without rust
6 randomSeed <- 3788179 #NA or a integer between 10.000 and 9.999.999
7
8 ##### La Fe 2014-2019 #####
9 initAndRunOneYear(s0_sim)
10 # After this first year,
11 # either x years can be run
12 # /// Run full simulation ///
13 runYears( $Meteo$year[nrow($Meteo)] - $Meteo$year[1]) # By default stepSemanalDynaCof: (1 week of
14

```

Contiene instrucciones en R para ejecutar tanto DynaCof como MiRoya a partir de datos meteorológicos de La Fe, Honduras.

Configurar la ruta de acceso a DynaCof (Set the Path)

Para utilizar el modelo DynaCof por primera vez, hay que definir el PATH. Para ello, en RStudio, hay que hacer clic en el menú "Session" → "Set Working Directory" → "To Source File Location":



En la consola, se puede comprobarlo usando `getwd()`

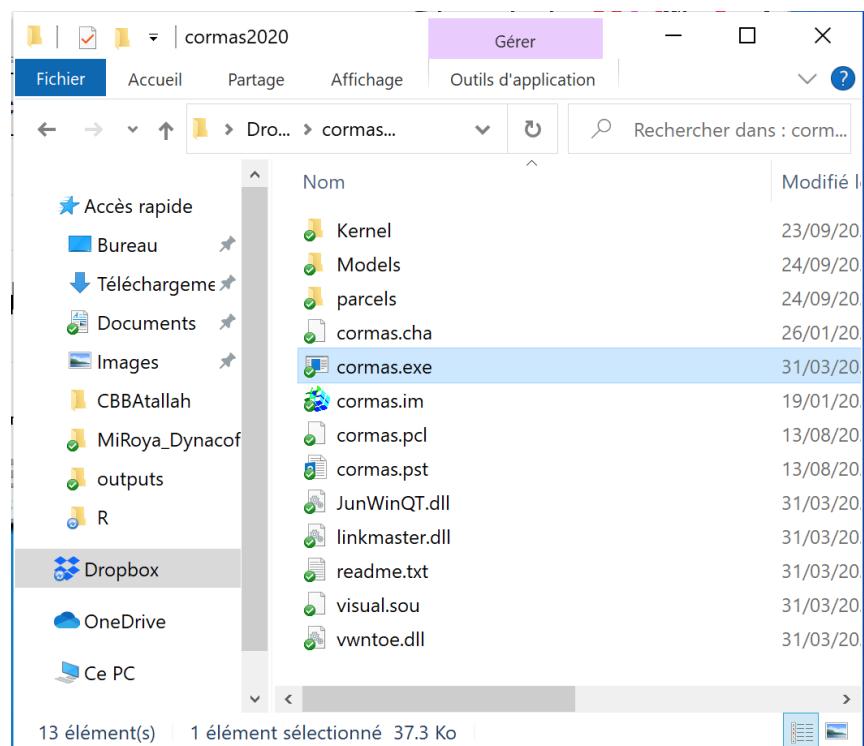
Felicitaciones, ambas plataformas y ambos modelos están instalados y listos para ser utilizados. Ahora para rodar el modelo, consulte la [Guía de usuario del modelo MiRoya.pdf](#).

Ejecutar MiRoya sin DynaCof

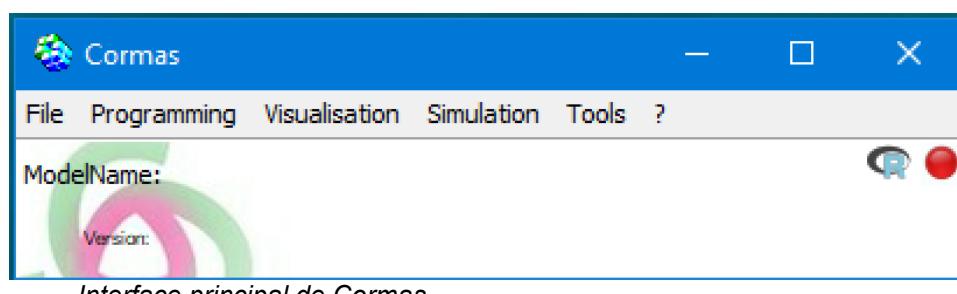
El modelo *MiRoya* ofrece una versión simplificada para simular la incidencia de la roya. En esta versión, el crecimiento del cafeto se describe de forma aproximada, pero nuestros análisis han demostrado que la dinámica de crecimiento del cafeto juega un papel importante en la dinámica general. Esta versión de *MiRoya* sin *DynaCof* es, por tanto, una versión degradada del modelo, pero es más rápida y fácil de ejecutar. *MiRoya* es ejecutable en la plataforma Cormas (<http://cormas.cirad.fr>).

Abrir Cormas

Después de instalar Cormas, haga doble clic en el archivo **cormas.exe**:

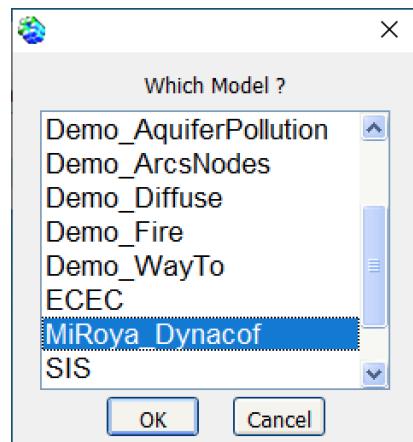


Nota: esta instalación contiene **cormas.exe** para Windows. También funciona en Mac y Linux, pero en este caso es necesario abrir el archivo **cormas.im** asociándolo a **cormas/macx/visual.app**

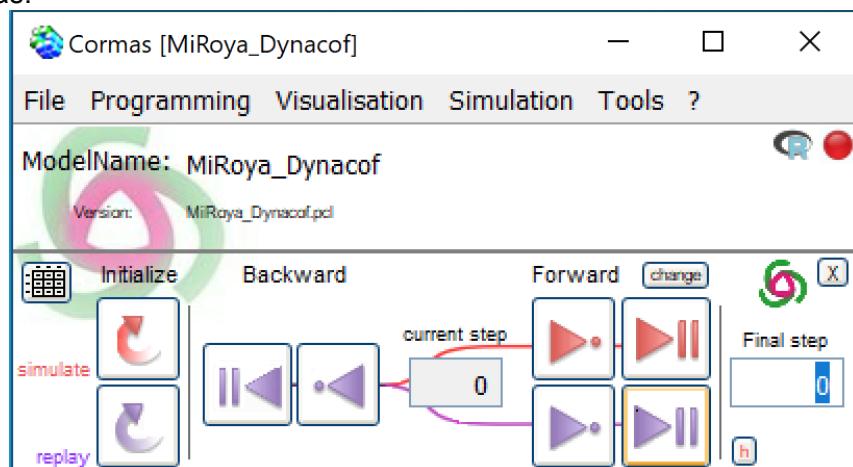


Cargar el modelo MiRoya

Se trata de probar una versión sencilla del modelo MiRoya sin acoplamiento con DynaCof. Desde la interfaz principal de Cormas, ir al menú "File" y luego "Load". Desde allí, seleccione MiRoya_Dynacof:



Luego seleccione el archivo **MiRoya_Dynacofpcl** y haga clic en OK. El modelo se carga así en Cormas:



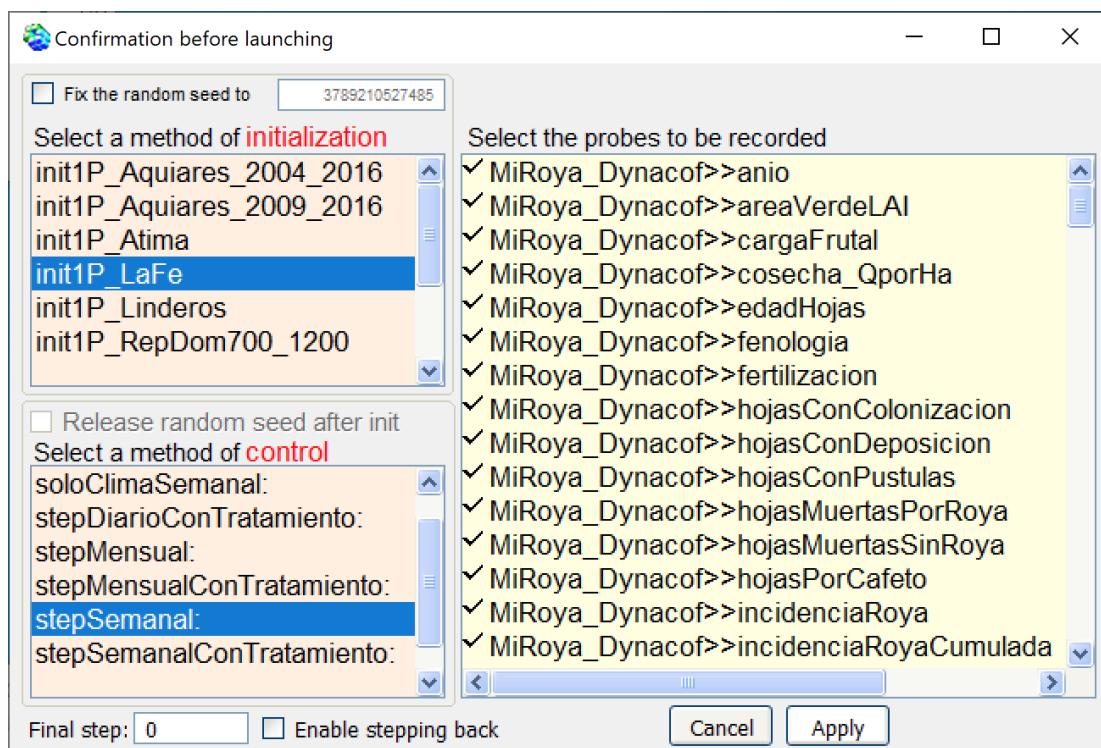
Probar MiRoya con Cormas

Configurar un estado inicial

Para simular MiRoya, primero debe crear un primer estado del "mundo". Esto se hace haciendo clic en el botón rojo llamado "**simulate**" :



Se abre una nueva interfaz de inicialización:



Interfaz para la construcción de un escenario de simulación en Cormas

En esta interfaz se debe seleccionar el método de inicialización (por ejemplo, *init1P_LaFe*) y el método de control *stepSemanal*. También tiene que seleccionar los indicadores que queremos seguir durante una simulación. Por lo tanto, deben ser seleccionados en el lado derecho de la interfaz (zona amarilla). Para seleccionarlos todos, haz clic en el primero y luego en el último con la tecla shift pulsada. Luego haga clic en "Apply".

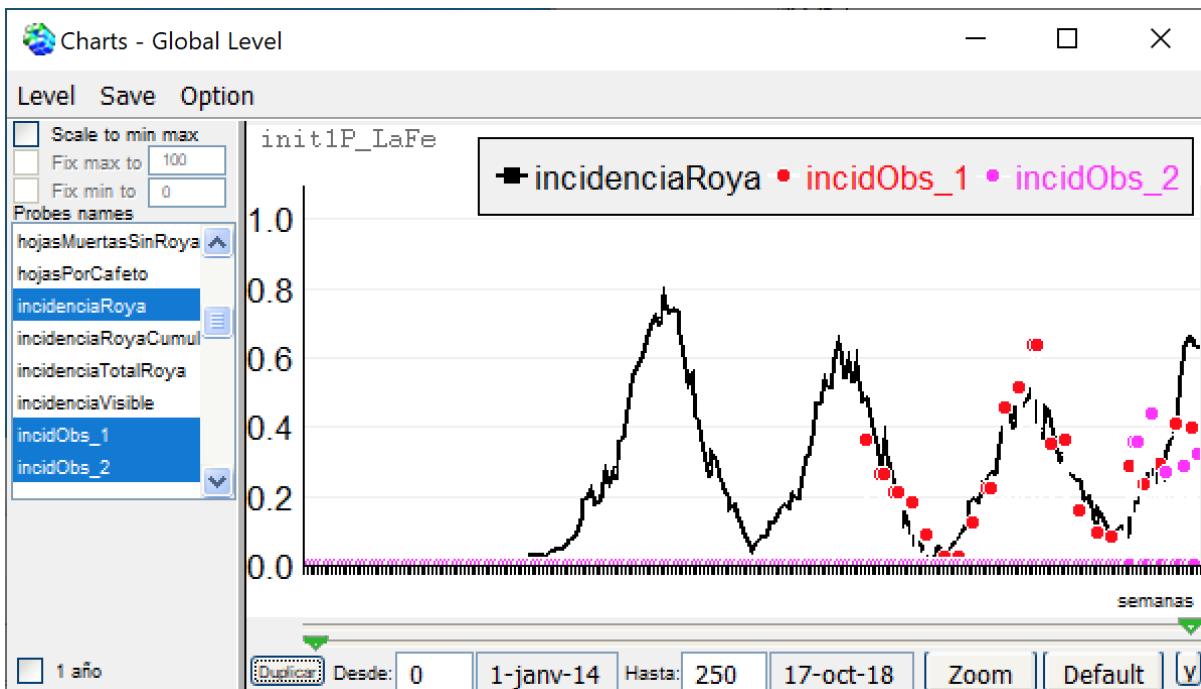
Simular 3 años

Cada clic en el botón rojo "Forward" hace que se ejecute una semana (un step) de simulación. Si queremos simular 3 años, ¡tendríamos que hacer clic 156 veces! En cambio, para correr 156 semanas seguidas, es mejor introducir el valor final (156 en el "Final step") y luego hacer clic en el botón rojo de avance rápido :



Ver los indicadores

Para ver los indicadores (también llamados sondas o probes en inglés), ir al menú "Visualisation" -> "Probes". A continuación, seleccione los indicadores de su elección:



Interfaz de los indicadores: la curva negra representa la incidencia simulada por el modelo, los puntos rojos y rosas corresponden a incidencias observadas en La Fe.

Para ver varios indicadores al mismo tiempo, utilice la tecla CTRL.

Ejecutar Dynacof sin MiRoya

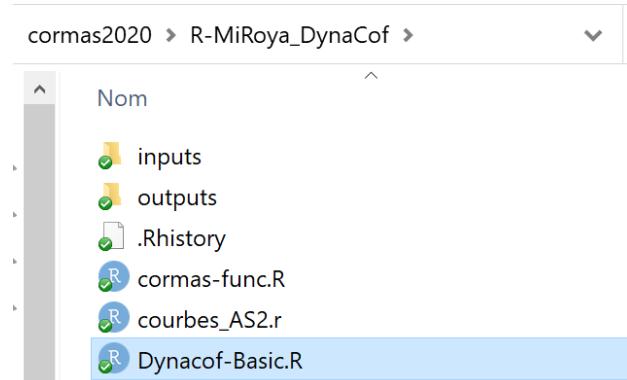
Esta sección explica cómo ejecutar el modelo DynaCof solo en R. DynaCof simula el crecimiento de un cafeto sin roya.

Simulación por defecto

Por defecto, DynaCof puede realizar simulaciones basadas a partir de datos meteorológicos de Aquiares (Costa Rica) de 1982 a 1990. Para ello, en RStudio, escriba las siguientes instrucciones y ejecútelas haciendo clic en el botón "Run" (después de seleccionar estas instrucciones):

```
#remotes::install_github("VEZY/DynACof")
rm(list = ls())
library("DynACof")
Sys.setenv(TZ="UTC")
sim = DynACof(Period= as.POSIXct(c("1979-01-01", "1985-12-31")))
```

Estas instrucciones también están en el archivo *Dynacof-Basic.R* en C:\...\cormas2020\R-MiRoya_DynaCof\ :



La siguiente captura de pantalla muestra el progreso de una simulación básica de DynaCof con datos meteorológicos de 1982 a 1990:

```
> S=DynACof(Period= as.POSIXct(c("1982-01-01", "1990-12-31")))
Meteo computation done
```

Meteo computation done

```
Starting a simulation from 1982-01-01 to 1989-12-31 over 1 plantation cycle(s)
=====
```

| 33%

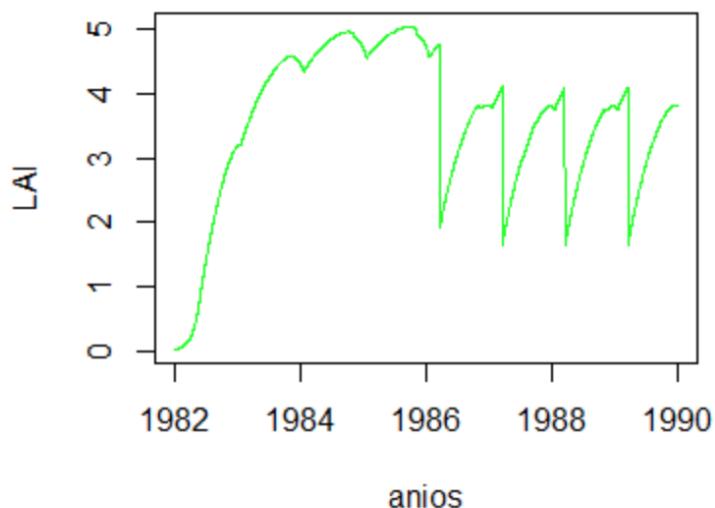
Los datos de esta simulación se almacenan en un `data.frame` llamado “**sim**”.

Escriba y ejecute la siguiente instrucción para mostrar un gráfico de LAI (Leaf Area Index):

```
plot(sim$Meteo$date,sim$Sim$LAI)
```

o, de forma más limpia:

```
plot(sim$Sim$LAI~sim$Meteo$date,type="l",col="green",lwd=1,xlab="anios",ylab="LAI")
```



Este gráfico muestra la evolución de la superficie foliar (LAI) a lo largo del tiempo. En los primeros años, el crecimiento es continuo: se parece un poco a una curva logística que alcanza un máximo. A partir del quinto año, se realiza una poda del 60% cada año (de ahí la fuerte disminución de LAI). Esta información de poda se define en el archivo de configuración `coffeePruning.R` que se encuentra en el subdirectorio “inputs”. Al editar este archivo, se puede leer que la edad de la primera poda está establecida a los 5 años (`MeanAgePruning = 5`), que la tasa de poda es del 60% (`LeafPruningRate = 0.6`) y que el día del año en que se realiza la poda (`D_pruning = 74`) está definido en el día 74 o sea el 16 de marzo:

```

1 #' @rdname site
2 #' @export
3 coffee= function(){
4   list(
5     Stocking_Coffee = 5580,      # Coffee density at planting (plant ha-1)
6     AgeCoffeeMin = 1,           # minimum coffee stand age
7     AgeCoffeeMax = 41,          # maximum coffee stand age (start a new rotation after)
8     SLA = 10.97,               # Specific Leaf Area (m-2 kg-1 dry mass)
9     wleaf = 0.068,              # Leaf width (m)
10    DELM = 2.0,                # Max Leaf carbon demand (gC plant-1 d-1)
11    Height_Coffee = 2,          # Average coffee canopy height (m), used for aerodynamic conductance.
12    D_pruning = 74,             # day of year of pruning
13    MeanAgePruning = 5,         # Age of first pruning (year)
14    LeafPruningRate = 0.6,       # how much leaves are pruned (ratio)
15    WoodPruningRate = 1/3,        # how much branches wood are pruned (ratio)
16    k_Dif = 0.3905968,          # Light extinction coefficient for diffuse light (-), computed from MAESPA
17    k_dir = 0.3109511           # light extinction coefficient for direct light (-) computed from MAESPA
18

```

Archivo de configuración *coffeePruning.R*

Ejecutar una simulación sin poda

Para ejecutar *DynaCof* con datos específicos, es necesario configurar el modelo. Aquí, por ejemplo, queremos simular a partir de datos meteorológicos de Aquiares, sin podar.

Configurar el tipo de manejo

Para eliminar la poda, editar el archivo *coffeePruning.R*⁴ y poner los valores de

LeafPruningRate y *WoodPruningRate* a 0:

```

LeafPruningRate = 0, #0.6,           # how much leaves are pruned (ratio)
WoodPruningRate = 0, #1/3,            # how much branches wood are pruned (ratio)

```

Sin embargo, esta información ya está definida en el archivo *CoffeeNoPruning.R* (que será llamado al ejecutar el método *DynaCof()*).

Comparar dos simulaciones

Aquí, para simular *DynaCof*, tomamos datos de La Fe, Honduras.

El archivo climático llamado *meteorology_LaFe_2014-2020.txt* se encuentra en la carpeta "inputs". Su propio archivo de datos debe tener el mismo formato. Si faltan los datos de radiación solar, puede copiarlos a partir de *meteorology_Aquiares.txt*.

```

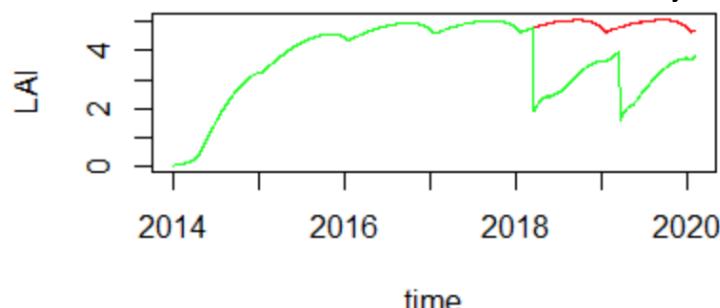
rm(list = ls())
library(DynACof)
#*****
basicName <- "LaFe_2014-2020"
#*****
fileNameMETEO <- paste("meteorology_", basicName, ".txt", sep = "")
Sys.setenv(TZ="UTC")
# Run a simulation without pruning
sim <- DynACof(Period= as.POSIXct(c("2014-01-01", "2020-12-31")),
               Inpath = "inputs",
               FileName = list(Site = "site.R", Meteo =fileNameMETEO,
                               Soil = "soil.R", Coffee = "coffeeNoPruning.R"))
# Run a 2nd simulation with pruning
sim2 <- DynACof(Period= as.POSIXct(c("2014-01-01", "2020-12-31")),
                 Inpath = "inputs",
                 FileName = list(Site = "site.R", Meteo =fileNameMETEO,
                                 Soil = "soil.R", Coffee = "coffeePruning.R"))
#plot LAI of 2 simulations
plot(sim$Sim$LAI~sim$Meteo$Date,type="l",col="green",lwd=1,xlab="anios",ylab="LAI")
lines(sim2$Sim$LAI~ sim2$Meteo$Date,col="green")

```

(cf. *DynacofOnly-main_LaFe.R*)

⁴ Disponible también en: <https://github.com/VEZY/DynACof/blob/master/R/0-Coffee.R>

Esto simula dos veces 6 años de crecimiento del cafeto en La Fe, sin y con poda:



Aquí, la función `DynACof()` ejecuta la simulación completa durante el periodo seleccionado. También es posible simular día por día utilizando el método `dynacof_i()`. Esta función se utiliza para el acoplamiento con *MiRoya*.

Configurar los inputs meteorológicos

El archivo `1-Meteo.R`⁵ describe los parámetros climáticos necesarios para ejecutar el modelo. La siguiente tabla presenta estos parámetros y especifica si son indispensables o pueden ser calculados por el modelo:

Tabla 1. de variables meteorológicas necesarias para DynaCof

| Variable | Definition | Required or not |
|-----------------|---|--|
| Date | Date in POSIXct format | *If missing* Computed from start date parameter, or set a dummy date |
| year | Year of the simulation | *If missing* Computed from Date |
| DOY | day of the year | *If missing* Computed from Date |
| Rain | Rainfall (mm/day) | Required (otherwise error) |
| Tair | Air temperature (Celsius above canopy) | *If missing* Computed from Tmax and Tmin |
| Tmax | Maximum air temperature during the day (Celsius) | Required (otherwise error) |
| Tmin | Minimum air temperature during the day (Celsius) | Required (otherwise error) |
| RH | % of Relative humidity | Not used but preferred over VPD for Rn computation |
| RAD | Incident shortwave radiation (MJ m-2 d-1) | *If missing* Computed from PAR |
| Pressure | Atmospheric pressure (hPa) | *If missing* Computed from VPD, Tair and Elevation or alternatively from Tair and Elevation. |
| WindSpeed | Wind speed (m s-1) | *If missing* Taken as constant: `Parameters\$WindSpeed` |
| CO2 | Atmospheric CO2 concentration (ppm) | *If missing* Taken as constant: `Parameters\$CO2` (374 ppm) |
| DegreeDays | Growing degree days (Celsius) | *If missing* Computed using [GDD()] |
| PAR | Incident photosynthetically active radiation (MJ m-2 d-1) | *If missing* Computed from RAD |
| FDiff | Diffuse light fraction | *If missing* Computed using [Diffuse_d()] using Spitters et al. (1986) formula |
| VPD | Vapor pressure deficit (hPa) | *If missing* Computed from RH |
| Rn | Net radiation (MJ m-2 d-1) | *If missing* Computed using [Rad_net()] with RH or VPD |
| DaysWithoutRain | Number of consecutive days with no rainfall | *If missing* Computed from Rain |
| Air_Density | Air density of moist air above canopy (kg m-3) | *If missing* Computed using |

⁵ Disponible en <https://github.com/VEZY/DynACof/blob/master/R/1-Meteo.R>

| | | |
|-----|---------------------------------------|---|
| | | [bigleaf::air.density()] |
| ZEN | Solar zenithal angle at noon (radian) | *If missing* Computed from Date, Latitude, Longitude and Timezone |

Entonces, para ejecutar *DynaCof*, se necesitan **datos meteorológicos diarios** que contengan al menos información sobre :

- temperatura mínima del día = "Tmin"
- temperatura máxima del día = "Tmax".
- temperatura media del aire = "Tair"
- Precipitación del día = "Lluvia"
- Radiación solar = "RAD"
- Humedad relativa = "RH" (si no hay, se necesita la VPD: Déficit de presión de vapor)
- Velocidad del viento = "WindSpeed" (puede ser una constante = 1 m/s)
- Presión = "Pressure" (si no hay, se necesita la T° y altitud)

(En el capítulo [Convertir el archivo para MiRoya y DynaCof](#), se describe una forma de conseguir que este archivo se formatee automáticamente para DynaCof)

Ejemplo de archivo (formato CSV):

| | A | B | C | D | E | F | G | H |
|---|------------|---------------|------------------|--------|--------|----------------|----------------|-----------|
| 1 | Date | Precipitacion | TemperaturaMedia | RH | RAD | TemperaturaMAX | TemperaturaMIN | WindSpeed |
| 2 | 01/01/2009 | 5.18 | 16.152 | 93.74 | 5.385 | 17.31 | 14.88 | 1.32 |
| 3 | 02/01/2009 | 0 | 17.273 | 88.65 | 16.929 | 21.46 | 14.92 | 1.01 |
| 4 | 03/01/2009 | 1.531 | 17.755 | 87.354 | 18.625 | 21 | 14.14 | 1.34 |
| 5 | 04/01/2009 | 0.204 | 18.325 | 89.423 | 18.54 | 22.27 | 15.13 | 1.58 |
| 6 | 05/01/2009 | 37.712 | 18.08 | 94.85 | 7.556 | 20.46 | 16.3 | 1.86 |

Para ser utilizados por *DynaCof*, estos datos deben almacenarse en un archivo (al formato **txt**) llamado “**meteorology_Aquiares.txt**” en la carpeta C:\...\cormas2020\R-MiRoya_DynaCof\inputs\ :

 meteo_Aquiares - 2009-2016.txt - Bloc-notes

Fichier Edition Format Affichage Aide

```
'Date','Rain','Tair','RH','RAD','Tmax','Tmin','WindSpeed','year','doy',
"2009-01-01",5.18,16.152,93.74,5.385,17.31,14.88,1.32,2009,1,
"2009-01-02",0,17.273,88.65,16.929,21.46,14.92,1.01,2009,2,
"2009-01-03",1.531,17.755,87.354,18.625,21,14.14,1.34,2009,3,
"2009-01-04",0.204,18.325,89.423,18.54,22.27,15.13,1.58,2009,4,
```

Para generar este archivo con el formato de fecha correcto, se deben ejecutar las siguientes instrucciones:

```
# ** to convert the Dates as "2015-01-01" (in case you get a date format
error)
fileNameMETEO <- "meteorology_Aquiares.csv"
data <- read.table(paste(getwd(),"/inputs/",fileNameMETEO, sep = ""),
header = TRUE, sep = ";")
data[,3] <- as.Date(data[,3], format = "%d/%m/%Y")
write.table(data, file=paste(getwd(),"/inputs/",fileNameMETEO, ".txt", sep =
""), sep = ",", col.names = TRUE, row.names = FALSE)
```

Por último, ejecute el modelo utilizando sus archivos de parámetros personalizados:

```
rm(list = ls())
```

```

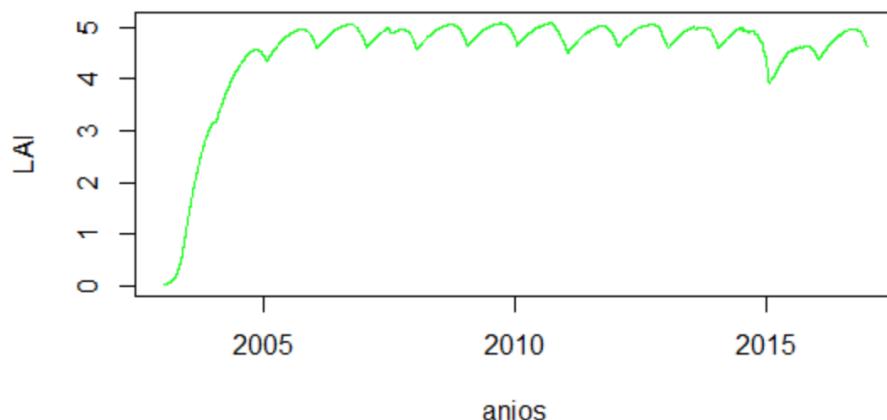
library(DynACof)
#*****basicName <- "Aquiares"
#*****fileNameMETEO <- paste("meteorology_", basicName, ".txt", sep = "")
Sys.setenv(TZ="UTC")
sim <- DynACof(Period= as.POSIXct(c("2003-01-01", "2016-12-31")),
                Inpath = "inputs",
                FileName = list(Site = "site.R", Meteo =fileNameMETEO,
                                Soil = "soil.R", Coffee = "coffeeNoPruning.R"))
plot(sim$Sim$LAI~sim$Meteo$Date, type="l", col="green", lwd=1, xlab="anios", yla
b="LAI")

```

Este script también está disponible en el archivo *DynacofOnly-main.R*.

Tenga en cuenta que la función *DynACof()* utiliza, por defecto, la siguiente lista de archivos de entrada: `list(Site="Site.R",Soil="Soil.R",Coffee="Coffee.R",Tree=NULL)`.

Aquí la simulación se ejecuta sin Poda ni árboles de cobertura (el parámetro Tree se pone a NULL). Como resultado, se obtiene:



Así, se puede ejecutar una simulación a partir de sus propios datos.

Ejecutar MiRoya-Dynacof con R y Cormas

Esta parte explica cómo ejecutar el modelo completo: DynaCof + MiRoya.

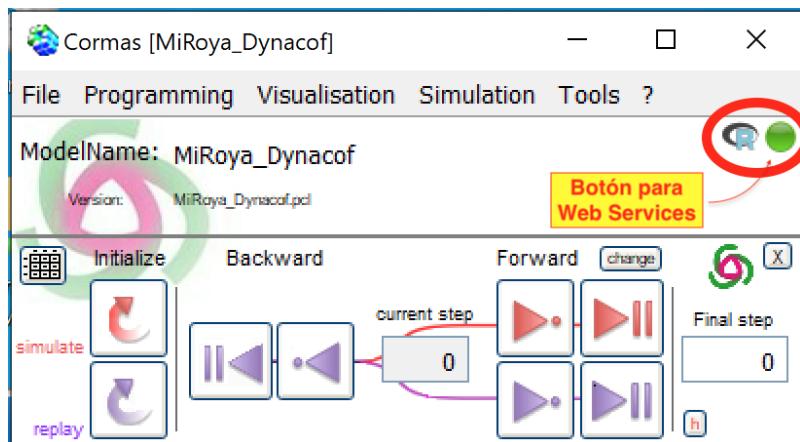
Acoplamiento ya configurado

Aquí presentamos cómo ejecutar un modelo ya configurado para Cormas y R: los archivos climáticos y de arranque ya están disponibles. Las partes siguientes explican cómo configurar estos archivos.

Activar el acoplamiento del lado Cormas

El modelo MiRoya-DynaCof se controla desde R. Cormas actúa entonces como un simulador secundario que responde a las solicitudes de servicio enviadas por R. Para ello, el **WebService** debe ser activado en Cormas.

Después de abrir Cormas y cargar *MiRoya_Dynacof* (como se explica en la [sección 3](#)), haga clic en el botón rojo ("Start/Stop WebServices") para activar los webServices. En este caso, el botón se vuelve verde.



Ahora, Cormas escucha las demandas de servicio provenientes de R y puede responder a ellas. Si se olvida de activar este botón, se producirá un error “`Connection refused`” al iniciar una simulación desde R:

Date init: 2014 -> Date fin: 2019

Error in curl::curl_fetch_memory(url, handle = handle) :
Failed to connect to localhost port 4920: Connection refused

Controlar el modelo del lado R

Para funcionar, el acoplamiento utiliza las funciones definidas en el archivo *func-MiRoya-DynaCof.R*. Por lo tanto, el script R debe recurrir a este archivo:

Este script está incluido en el archivo *main-MiRoya-Dyn_LAI_LaFe.R*.

Explicaciones de los tres parámetros :

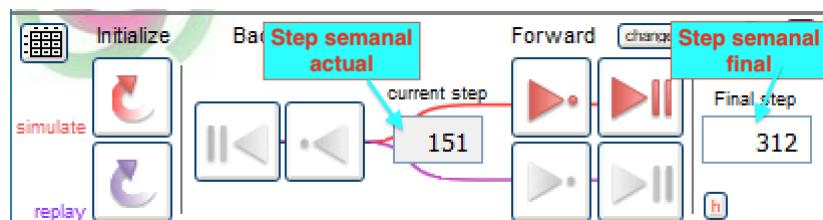
- `cormasInit <- "init1P_LaFe"` se utiliza para indicar a Cormas qué método de inicialización debe ejecutarse. Aquí por ejemplo, queremos simular una parcela con datos meteorológicos de La Fe, Honduras.
 - `S0_sim <- "S0_sim_LaFe_2014-2020.rda"` corresponde al archivo que contiene los resultados de una primera simulación **sin roya**. Este archivo es necesario para ejecutar Dynacof paso a paso (y se utiliza como arranque para la simulación). El siguiente párrafo explica cómo generar este archivo y dónde almacenarlo. Tenga en cuenta que el archivo `S0_sim_LaFe_2014-2020.rda` ya está disponible en la carpeta "outputs".
 - `randomSeed <- integer or NA` es el valor de la semilla aleatoria para MiRoya. En efecto, MiRoya es un modelo estocástico, es decir, utiliza funciones en las que intervienen suertes aleatorias. Una semilla aleatoria es un número utilizado para inicializar un generador de números pseudoaleatorios. La secuencia de números aleatorios producida por el generador deriva de forma determinista del valor de la semilla. Por otro lado, dos semillas diferentes producirán secuencias de números

aleatorios completamente diferentes. Si no se especifica la semilla (NA), se generará a partir del estado del sistema leyendo el valor del reloj.

En tiempo de ejecución, la consola R muestra los números de step diario y los tiempos de espera (para la comunicación con Cormas):

```
g 1.2 s... waiting 1.4 s... waiting 1.6 s... waiting 1.8 s... waiting 2 s... waiting 2.2
s... waiting 2.4 s... Step 555 OK.
waiting 0.2 s... waiting 0.4 s... waiting 0.6 s... waiting 0.8 s... waiting 1 s... waitin
g 1.2 s... waiting 1.4 s... waiting 1.6 s... waiting 1.8 s... waiting 2 s... waiting 2.2
s... waiting 2.4 s... Step 562 OK.
waiting 0.2 s... waiting 0.4 s... waiting 0.6 s... waiting 0.8 s... waiting 1 s...
```

En el lado de Cormas, los steps semanales se muestran en la interfaz principal :



El paso final se define automáticamente a partir de los datos meteorológicos que se cargan en la inicialización del modelo. Tenga en cuenta que para MiRoya, estos datos están disponibles en la carpeta \cormas2020\Models\MiRoya_Dynacof\data\, mientras que los mismos datos deben almacenarse en la carpeta \cormas2020\R-MiRoya_DynaCof\inputs\ para DynaCof. Por lo tanto, se trata de dos archivos meteorológicos que contienen la misma información, pero que se almacenan en dos lugares diferentes, con formatos distintos.

Seguimiento de los indicadores

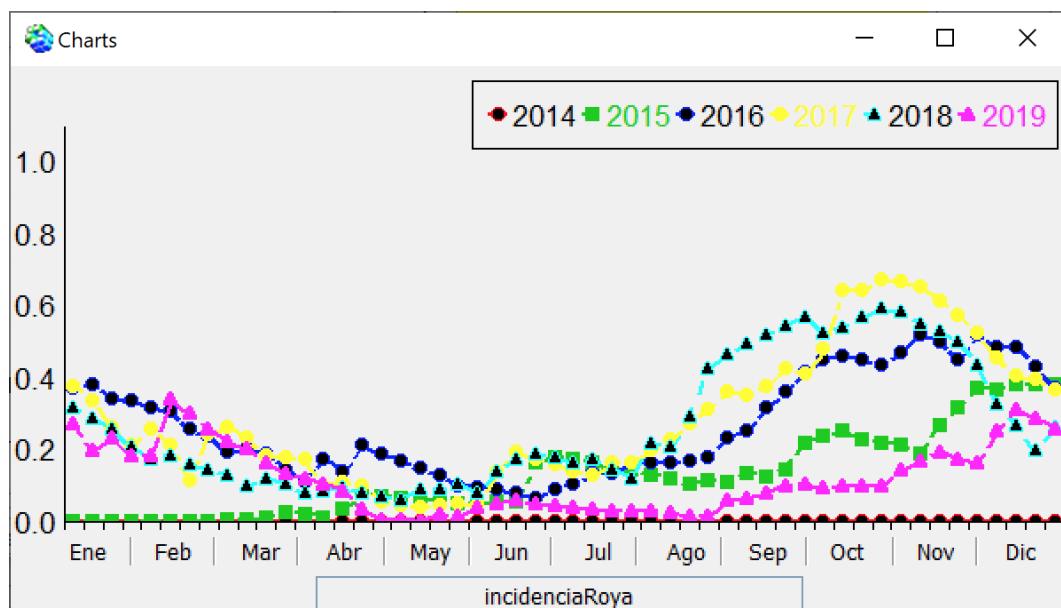
Las simulaciones son bastante lentas. Para seguir la evolución de un indicador, hay que abrir una interfaz gráfica. Para ello, en Cormas, seleccione el menú “Visualisation” → “Probes”. A continuación, haga clic, por ejemplo, en el indicador *areaVerdeLAI*:



Para visualizar varios indicadores al mismo tiempo, abra otra interfaz o seleccione otro indicador manteniendo pulsada la tecla CTRL.

Para ver un indicador en un solo año, puede

- o seleccionar la pequeña casilla llamada "1 año" en la parte inferior izquierda (para ver sólo el último año),
- o haga clic en el pequeño botón "y" (para "year") en la parte inferior derecha para ver la totalidad de la curva concentrada en los 12 meses:



Simulación de MiRoya-DynaCof con sus propios datos

Convertir los archivos de meteo para DynaCof

MiRoya ofrece dos algoritmos para transformar los datos meteorológicos de manera que puedan ser utilizados por el modelo. Así, puede utilizar sus propios datos de una estación meteorológica, o datos del sitio web de la NASA (ver próximo capítulo). Para sus propios datos, deben estar disponibles en un archivo CSV con esta disposición:

| | A | B | C | D | E | F | G | H |
|---|------------|-------|-------|-------|------|----------|-----------|-------|
| 1 | Date | Tair | Tmin | Tmax | Rain | Pressure | WindSpeed | RH |
| 2 | 01/01/2014 | 18.95 | 14.79 | 23.99 | 1.95 | 913.9 | 0.92 | 88.82 |
| 3 | 02/01/2014 | 19.66 | 16.44 | 24.12 | 1.32 | 913.7 | 0.84 | 87.53 |
| 4 | 03/01/2014 | 20.51 | 16.88 | 25.35 | 0.7 | 913.2 | 0.69 | 88.51 |

Pero, no hay un orden impuesto en las columnas de sus datos.

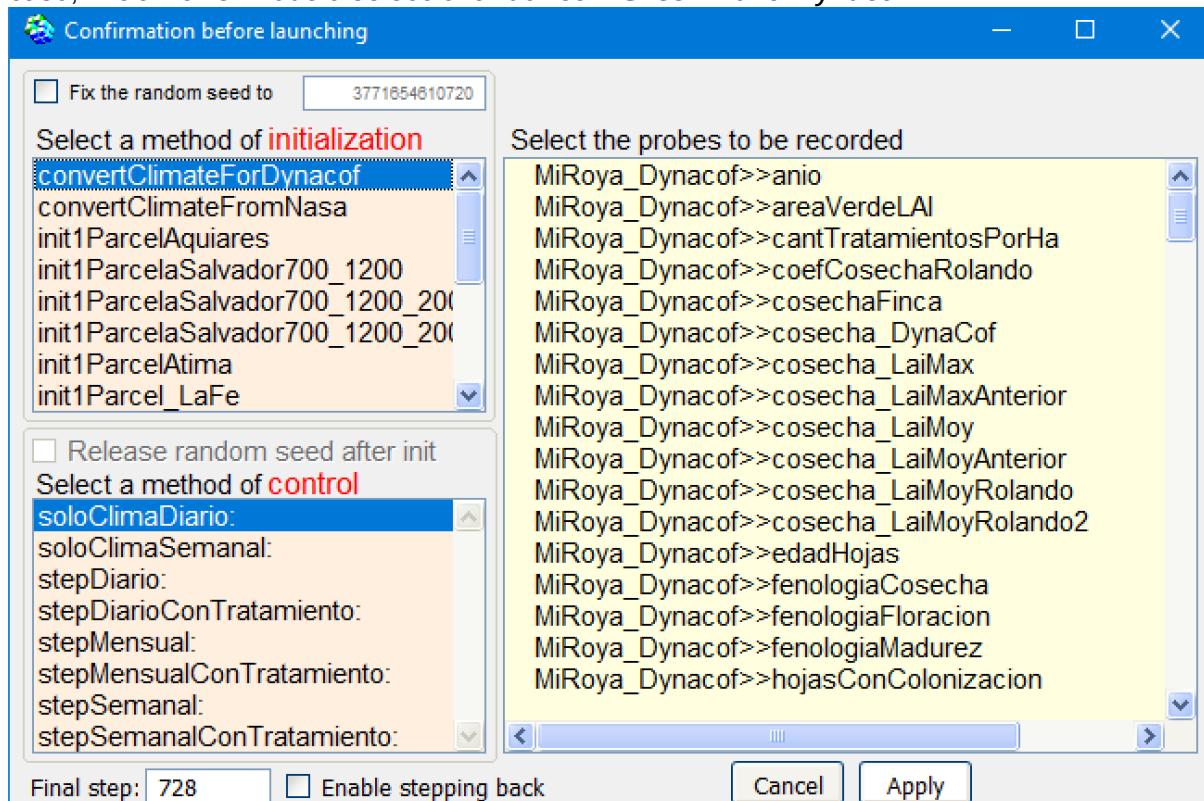
La siguiente lista presenta los nombres (Label) que pueden tomar sus variables. No hay requisitos de mayúsculas o minúsculas para los nombres de sus variables. Además, estos nombres pueden incluir espacios o guiones bajos (_):

- **Lluvia** = precipitacion, precipitation, lluvia o rain
- **Temperatura** = tair, temperaturamedia, meantemperature, daytemperature, dailytemperature, temperaturadeldía, temperaturadiaria, temperaturaaire, airtemperature.
- **Humedad relativa** = rh, hr, relativehumidity, humedadrelativa, humedad, humidity.
- **Radiación solar** = rad, radiation, radiaciónsolar, radiacionsolar, radiación, radiacion o solarradiation.
- **Presión** = presión, pressure o presion.
- **Velocidad del viento** = windspeed, ws, velocidaddelviento o velocidadviento.

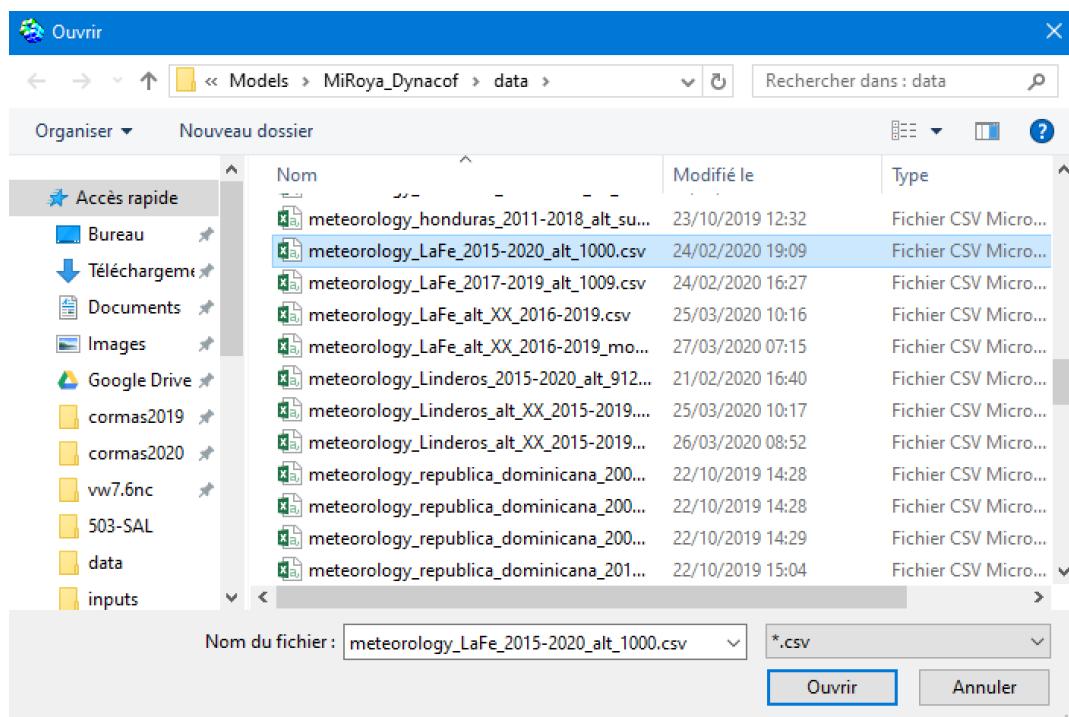
- **Temperatura máxima** = `tmax`, `temperaturamax`, `maximumtemperature`, `temperaturamáxima` o `temperaturamaxima`.
- **Temperatura mínima** = `tmin`, `temperaturamin`, `minimumtemperature`, `temperaturaminima` o `temperaturamínima`.

Así por ejemplo, para la temperatura media, se aceptan los siguientes nombres: `t_Air`, `temperaturamedia`, `temperatura media`, `Temperatura_media`, etc.

Para convertir estos datos en un nuevo archivo legible por DynaCof, utilice Cormas. En este caso, inicializar el modelo seleccionando “convertClimaForDynacof”.



Después del “Apply”, el usuario debe seleccionar su archivo CSV:



Si lo necesita, Cormas le propone añadir un año más: una copia del primer año (ejemplo: agregar el año 2014 si el primer año del archivo es 2015).

La [tabla 1](#) muestra las variables necesarias para DynaCof. Pero si no se dispone de las variables Presión, Humedad Relativa (HR) y Viento, el modelo MiRoya propone estimar estos datos. Así, la lista **mínima** de datos diarios debe contener:

- temperatura mínima del día = "Tmin"
- temperatura máxima del día = "Tmax".
- temperatura media del aire = "Tair"
- Precipitación del día = "Lluvia"
- Radiación solar = "RAD"

Tabla 2: lista mínima de variables

| | A | B | C | D | E | F |
|---|------------|---------------|------------------|--------|----------------|----------------|
| 1 | Date | Precipitacion | TemperaturaMedia | RAD | TemperaturaMAX | TemperaturaMIN |
| 2 | 01/01/2009 | 5.18 | 16.152 | 5.385 | 17.31 | 14.88 |
| 3 | 02/01/2009 | 0 | 17.273 | 16.929 | 21.46 | 14.92 |
| 4 | 03/01/2009 | 1.531 | 17.755 | 18.625 | 21 | 14.14 |
| 5 | 04/01/2009 | 0.204 | 18.325 | 18.54 | 22.27 | 15.13 |
| 6 | 05/01/2009 | 37.712 | 18.08 | 7.556 | 20.46 | 16.3 |

Los demás datos (RH, WindSpeed y Presión) se calculan y se almacenan en el nuevo archivo txt. En este caso:

- RH se estima mediante la fórmula:

Si la lluvia > 5mm entonces RH = valor aleatorio entre el 89% y el 100%.

Si no,

$$RH = 100 - 5 \cdot (tAir - Td)$$

Con: Td = dewpoint temperature (punto de rocío) :

$$Td = 0.38 \times Tmax - 0.018 \times Tmax^2 + 1.4 \times Tmin - 5$$

- Presión se estima mediante la altitud:

$$P = P_0 \times e^{-(\mu g h / R T)} \quad (P \text{ en HPa})$$

Con :

μ : Masa molar del aire terrestre, 0,0289644 kg/mol

g : Aceleración gravitacional, 9,80665 m/(s*s)

h : Diferencia de altitud, en metros

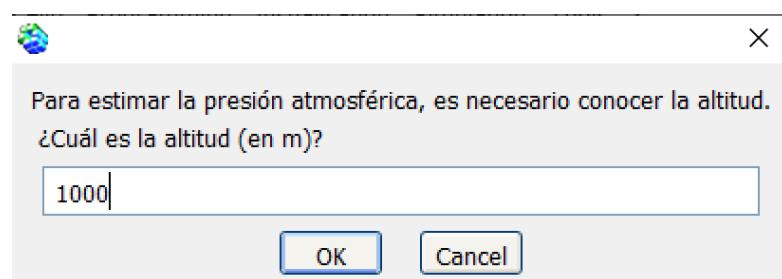
R : Constante universal de los gases para el aire, 8,31432 N·m /(mol·K)

T : Temperatura del aire, en K

El valor por defecto de P_0 es de 760 mmHg, que es la presión atmosférica estándar de 101,325 kPa o 1 atmósfera a 0 metros.

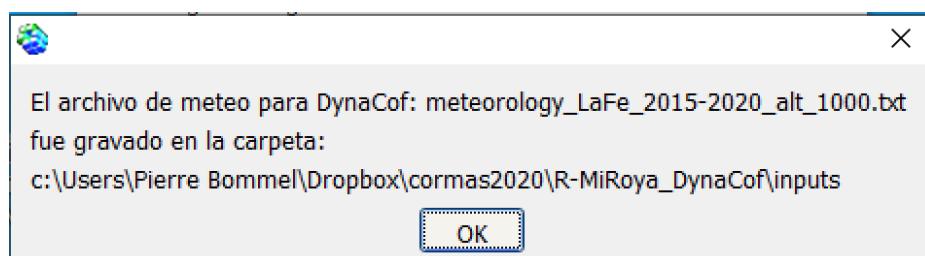
(1 mmHg = 133,3 Pascal = 1,333 HPa y K = °C + 273,15)

Nota que se pide al usuario que escriba la altitud:



- La velocidad del viento (WindSpeed) es constante: 1 m/s.

A final, un nuevo archivo al formato .TXT está grabado en la carpeta /R-MiRoya_DynaCof/inputs/ (ej: meteorology_LaFe_2015-2020_alt_1000.txt):



=> el archivo meteo está ahora listo para ser utilizado por DynaCof.

Utilizar datos de Clima de Nasa

Hay varios sitios web que permiten obtener datos meteorológicos. Aquí utilizamos como ejemplo el sitio de la NASA que proporciona gratuitamente datos diarios (son datos simulados): <https://photovoltaic-software.com/>

Un tutorial está disponible en inglés: [How to get solar radiation data anywhere in the world for free?](#)

El siguiente procedimiento explica cómo recuperar los datos y adaptarlos al modelo.

Recuperar los datos diarios de un punto GPS

Haga clic en “Power Data access Viewer : NASA solar radiation and meteorological data”. En el pequeño formulario “Power Single Point Data Access”, seleccionar:

1. “AgroClimatology” en el primer menú denominado “Choose a User Community”:

 POWER Single Point Data Access - X

1. Choose a User Community

Agroclimatology

2. Choose a Temporal Average

Daily Interannual Climatology

3. Enter Lat/Lon or Add a Point to Map

 14.989250 (-90 to +90 decimal degrees)

Clear -88.051573 (-180 to +180 decimal degrees)

4. Select Time Extent

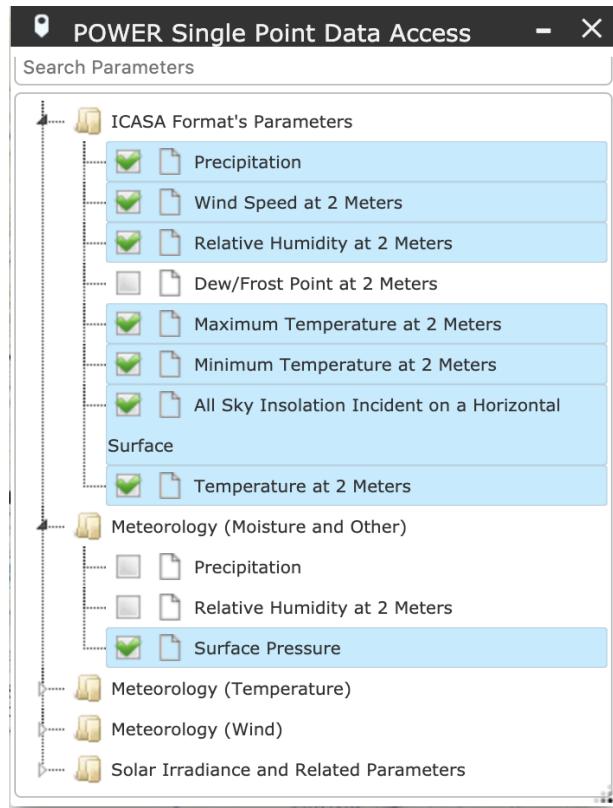
Start Date 01/01/2015 (MM/DD/YYYY)

End Date 02/01/2020 (MM/DD/YYYY)

5. Select Output File Formats Select All

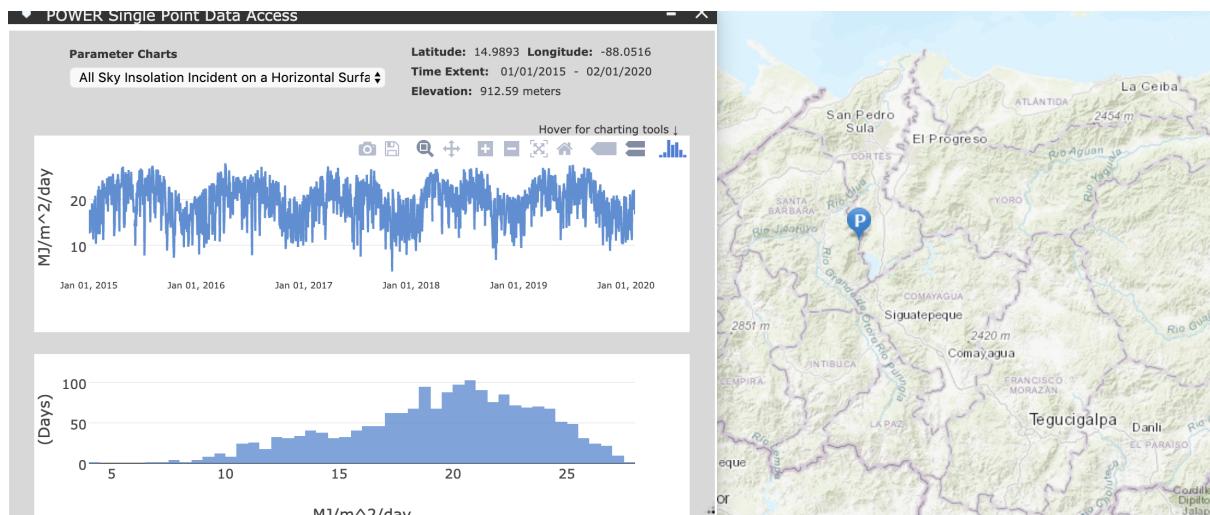
ASCII CSV GeoJSON ICASA NetCDF

2. Seleccionar **Daily** (para obtener datos diarios)
3. Indique el punto el punto **GPS** (o haga clic en el mapa)
4. Seleccionar la **duración**
5. Seleccionar el formato de archivo: **CSV**
6. Seleccionar **Select Parameters** para elegir los 8 datos climatológicos necesarios para el modelo : “[precipitation](#)”, “[Wind Speed at 2 meters](#)”, “[Relative Humidity at 2 Meters](#)”, “[Maximum Temperature at 2 Meters](#)”, “[Minimum Temperature at 2 Meters](#)”, “[All Sky Insolation on a Horizontal Surface](#)”, “[Temperature at 2 meters](#)” y “[Surface Pressure](#)”:



7. Finalmente, haga clic en el botón “**Submit**”.

A continuación, el sitio muestra los resultados de estos datos en forma de gráficos:

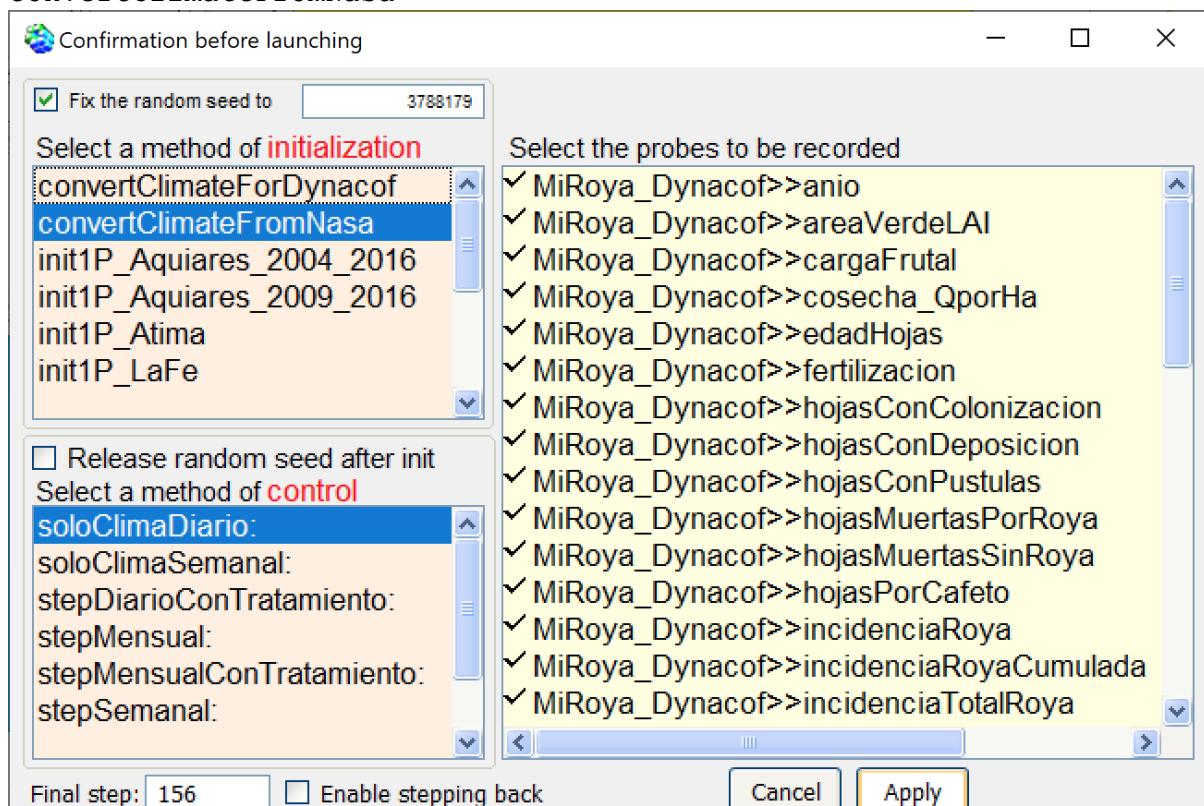


Después, debe hacer clic en el botón “**CSV**” para guardar estos resultados en un archivo CSV en la carpeta \cormas2020\Models\MiRoya_Dynacof\data\Nasa\.
Este archivo está de esta forma :

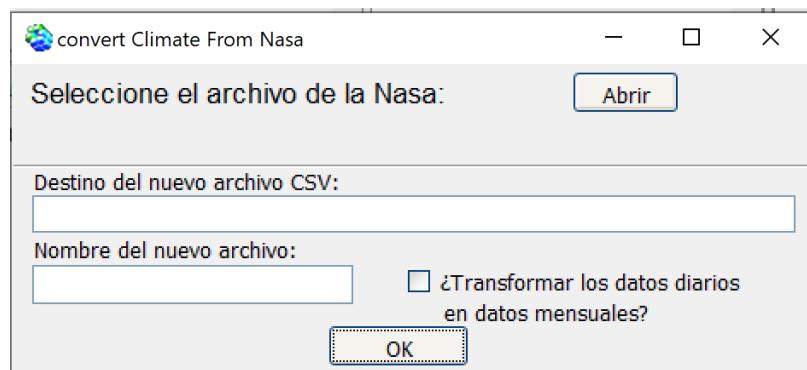
| | A | B | C | D | E | F | G | H | I | J | K | L |
|----|---|-----------|------|-----|---------|------|-------|---------|---------|----------------|-------|-------|
| 1 | -BEGIN HEADER- | | | | | | | | | | | |
| 2 | NASA/POWER SRB/FLASHFlux/MERRA2/GEOS 5.12.4 (FP-IT) 0.5 x 0.5 Degree Daily Averaged Data | | | | | | | | | | | |
| 3 | Dates (month/day/year): 01/01/2013 through 03/05/2015 | | | | | | | | | | | |
| 4 | Location: Latitude 14.4907 Longitude -88.1023 | | | | | | | | | | | |
| 5 | Elevation from MERRA-2: Average for 1/2x1/2 degree lat/lon region = 989.87 meters Site = na | | | | | | | | | | | |
| 6 | Climate zone: na (reference Briggs et al: http://www.energycodes.gov) | | | | | | | | | | | |
| 7 | Value for missing model data cannot be computed or out of model availability range: -99 | | | | | | | | | | | |
| 8 | Parameter(s): | | | | | | | | | | | |
| 9 | T2M_MIN MERRA2 1/2x1/2 Minimum Temperature at 2 Meters (C) | | | | | | | | | | | |
| 10 | RH2M MERRA2 1/2x1/2 Relative Humidity at 2 Meters (%) | | | | | | | | | | | |
| 11 | T2M MERRA2 1/2x1/2 Temperature at 2 Meters (C) | | | | | | | | | | | |
| 12 | WS2M MERRA2 1/2x1/2 Wind Speed at 2 Meters (m/s) | | | | | | | | | | | |
| 13 | PS MERRA2 1/2x1/2 Surface Pressure (kPa) | | | | | | | | | | | |
| 14 | PRECTOT MERRA2 1/2x1/2 Precipitation (mm day-1) | | | | | | | | | | | |
| 15 | ALLSKY_SFC_SW_DWN SRB/FLASHFlux 1/2x1/2 All Sky Insolation Incident on a Horizontal Surface (MJ/m^2/day) | | | | | | | | | | | |
| 16 | T2M_MAX MERRA2 1/2x1/2 Maximum Temperature at 2 Meters (C) | | | | | | | | | | | |
| 17 | -END HEADER- | | | | | | | | | | | |
| 18 | LAT | LON | YEAR | DOY | PRECTOT | WS2M | RH2M | T2M_MAX | T2M_MIN | ALLSKY_SFC_T2M | PS | |
| 19 | 14.49071 | -88.10229 | 2013 | 1 | 0 | 1.32 | 74.72 | 27.11 | 16.22 | 15.76 | 20.89 | 90.39 |
| 20 | 14.49071 | -88.10229 | 2013 | 2 | 0.01 | 1.01 | 70.77 | 27.62 | 16.53 | 19.18 | 21.04 | 90.33 |

Convertir el archivo para MiRoya y DynaCof

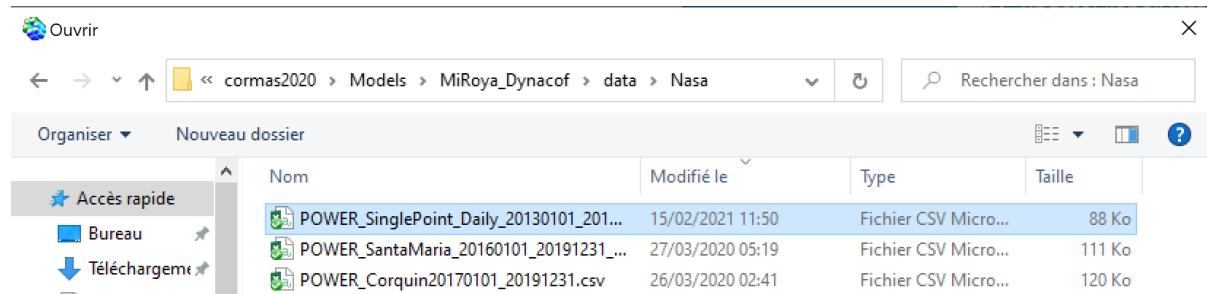
MiRoya ofrece un script para convertir estos datos en 2 archivos compatibles para MiRoya y DynaCof. Para ello, en Cormas, hay que seleccionar el método de inicialización convertClimateFromNasa



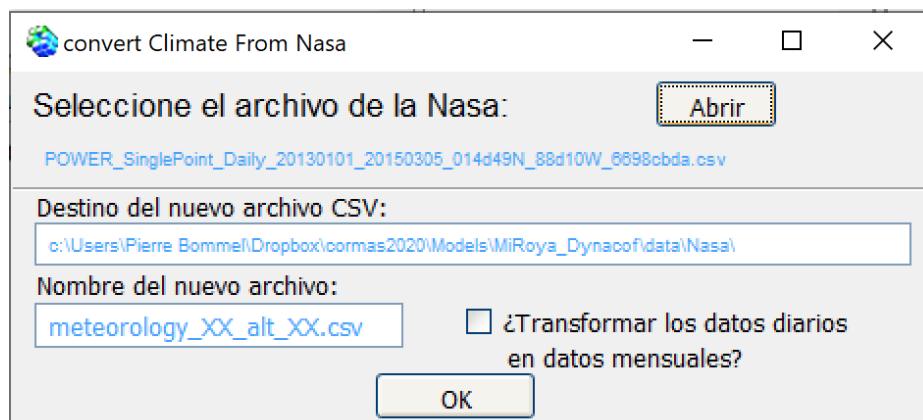
Después del "Apply", se abre una nueva ventana:



Debe hacer clic en el primer botón "Abrir" para seleccionar el archivo CSV previamente almacenado desde el sitio web de la NASA:

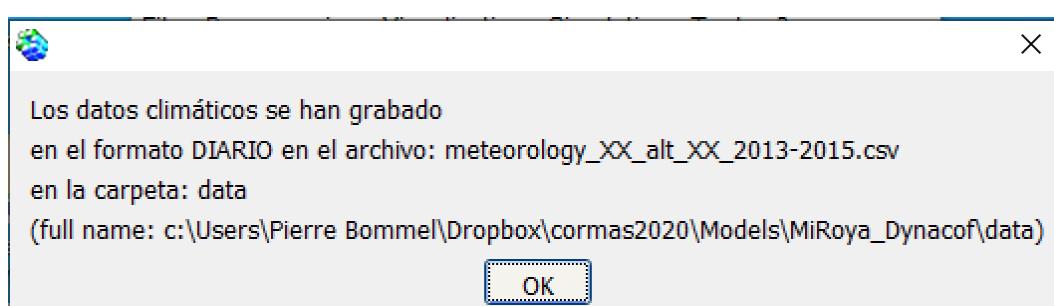


Para este ejemplo, mantenemos el nombre de archivo por defecto (meteorology_XX_alt_XX.csv). Por supuesto, puede elegir el nombre que desee. La interfaz debería tener el siguiente aspecto:

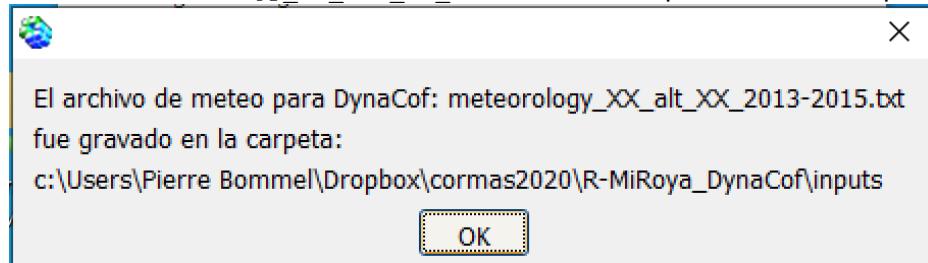


Después de hacer clic en el botón "OK", Cormas indicará las ubicaciones donde se almacenan los dos nuevos archivos meteorológicos:

- el archivo meteorology_XX_alt_XX_2013-2015.csv que será utilizado por MiRoya:



- y el archivo `meteorology_XX_alt_XX_2013-2015.txt` que será utilizado por DynaCof



Como se explica en la sección [Configurar los inputs](#) meteorológicos, este segundo archivo txt debe estar grabado en la carpeta `\cormas2020\R-MiRoya_DynaCof\inputs\` para estar utilizado por DynaCof.

El nuevo archivo (`meteorología_XX_alt_XX_2013-2015.csv`) creado por MiRoya, tiene entonces la siguiente forma :

| A | B | C | D | E | F | G | H | I | J | K | |
|---|------------|-------|-------|------|-----|-------|------|----------|-----------|-------|-------|
| 1 | Date | RH | Tmin | year | DOY | RAD | Rain | Pressure | WindSpeed | Tmax | Tair |
| 2 | 01/01/2013 | 74.72 | 16.22 | 2013 | 1 | 15.76 | 0 | 903.9 | 1.32 | 27.11 | 20.89 |
| 3 | 02/01/2013 | 70.77 | 16.53 | 2013 | 2 | 19.18 | 0.01 | 903.3 | 1.01 | 27.62 | 21.04 |
| 4 | 03/01/2013 | 73.86 | 16.42 | 2013 | 3 | 17.87 | 0.02 | 904.2 | 1.34 | 27.53 | 21.21 |
| 5 | 04/01/2013 | 70.68 | 17.51 | 2013 | 4 | 17.94 | 0.33 | 905.2 | 1.58 | 26.96 | 21.1 |

mientras que el archivo `meteorología_XX_alt_XX_2013-2015.txt` es de la forma :

```

meteorology_XX_alt_XX_2013-2015.txt - Bloc-notes
Fichier Edition Format Affichage Aide
>Date","RH","Tmin","year","DOY","RAD","Rain","Pressure","WindSpeed","Tmax","Tair","X"
"2013-01-01",74.72,16.22,2013,1,15.76,0,903.9,1.32,27.11,20.89,NA
"2013-01-02",70.77,16.53,2013,2,19.18,0.01,903.3,1.01,27.62,21.04,NA
"2013-01-03",73.86,16.42,2013,3,17.87,0.02,904.2,1.34,27.53,21.21,NA
"2013-01-04",70.68,17.51,2013,4,17.94,0.33,905.2,1.58,26.96,21.1,NA

```

Configurar MiRoya para utilizar el archivo CSV

Para utilizar este archivo de datos con *MiRoya*, se debe seleccionar el método de inicialización `init1P_XX` (ver [Ejecutar MiRoya sin DynaCof](#)).

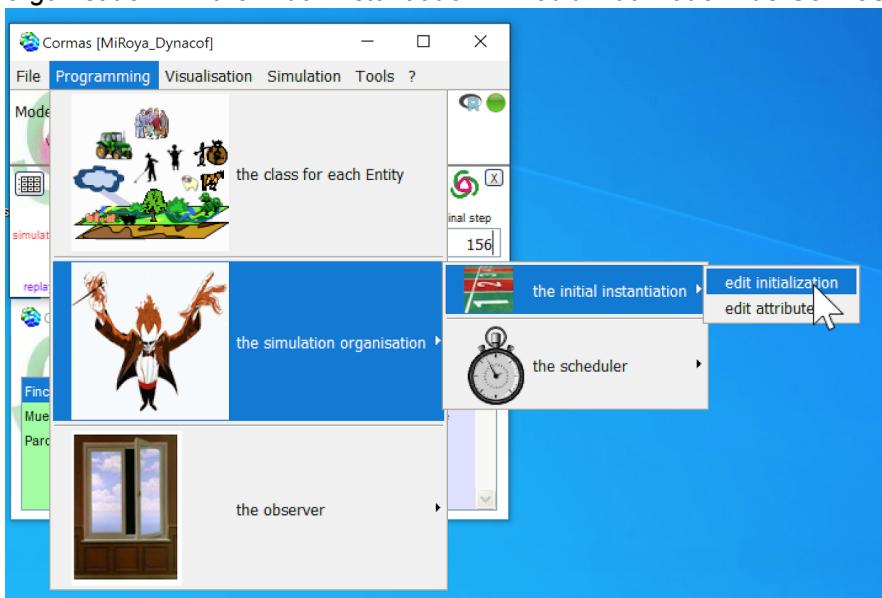
El código de este método contiene el nombre de este archivo:

The screenshot shows the MiRoya-DynaCof software interface. The top menu bar includes Browser, Edit, Find, View, Package, Class, Protocol, Method, Tools, and Help. The toolbar below has various icons for file operations. The main window has tabs for Package, Hierarchy, Instance, Class, Shared Variable, and Instance Variable. The 'Instance' tab is selected, showing a list of methods: controlling, converting, default values, init (selected), initOld, and init_ClimaMarkSim. Below this is a source code editor with the following content:

```
| meteoFile altitud subDirectory epidFile1 epidFile2 |
meteoFile := 'meteorology_XX_alt_XX_2013-2015.csv'.
subDirectory := nil.
epidFile1 := nil.
epidFile2 := nil.
altitud := 1001.
self init1ParcelMeteo: meteoFile directory: subDirectory epidemi1: epidFile1 epidemi2: epidFile2 altitud: altitud.
```

At the bottom, there are buttons for Method: #init1P_XX (init) and Package: MiRoya_Dynacof.

Para editar este método de inicialización, debe ir al menú “*Programming*” → “*the simulation organisation*” → “*the initial instantiation*” → “*edit initialization*” de Cormas:



Hacer la simulación S0 en DynaCof

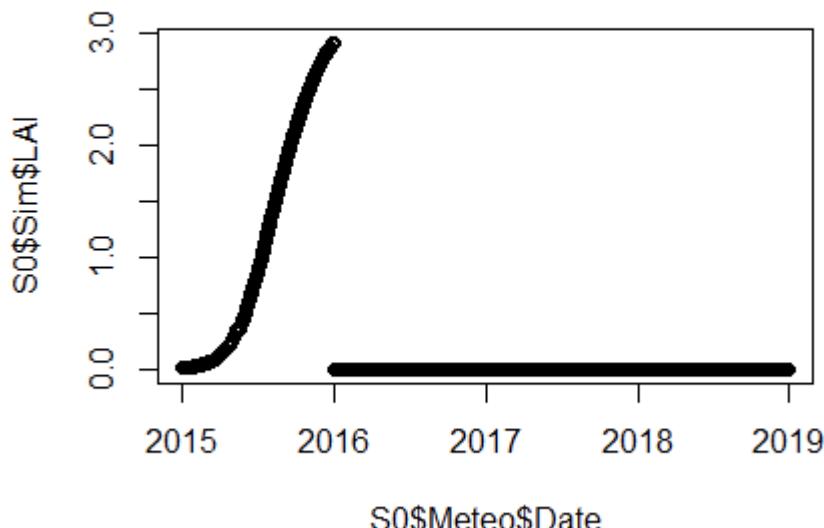
Para el acoplamiento MiRoya-DynaCof, debe realizarse previamente una simulación de arranque. Para ello, hay que realizar una simulación S0 de DynaCof. En R, ejecutar el archivo “DynaCof-sets0.R” o el siguiente script:

```

library(DynACof)
library(lubridate)
rm(list = ls())
# **** Parameters ****
meteo_fileName <- "meteorology_XX_alt_XX_2013-2015.txt"
dateInit <- "2013-01-01"
dateFin <- "2015-12-31"
pruning = "CoffeeNoPruning.r"
S0_name <- "S0_XX_2013-2015.rda"
# ----- ****
Sys.setenv(TZ="UTC")
# Convert meteo file for Dynacof
aFile <- paste(getwd(), "/inputs/", meteo_fileName, sep = "")
meteo <- read.table(aFile, header= TRUE, sep=",")
for (k in 2:length(meteo)){
  meteo[,k] <- as.numeric(meteo[,k])
}
write.csv(meteo, file = aFile, row.names = FALSE)

#Simulate just 1 year
S0 = dynacof_i(i = 1:365, Period= as.POSIXct(c(dateInit, dateFin)),
  Inpath = "inputs",
  FileName = list(Site = "Site.R",
    Meteo = meteo_fileName,
    Soil = "Soil.R",
    Coffee = pruning,
    Tree = NULL))
plot(S0$Meteo$Date, S0$Sim$LAI)

```



Esta simulación debe salvarse en la carpeta “outputs” para futuras simulaciones acopladas:

```
save(S0, file=paste(getwd(), "/outputs/", S0_name, sep = ""))
```

El archivo `S0_XX_2013-2015.rda` se utilizará para las siguientes simulaciones acopladas.

Hacer una simulación completa DynaCof-MiRoya

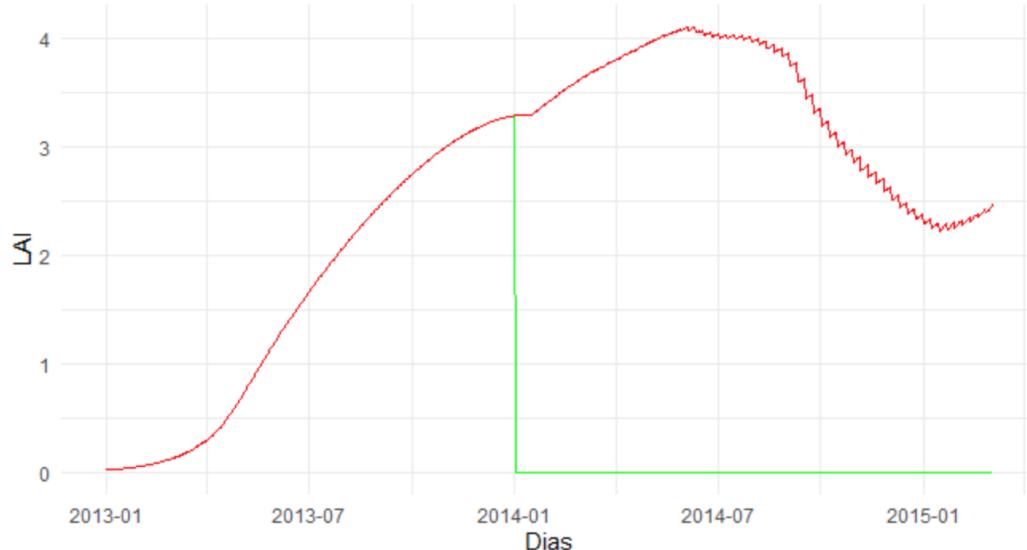
Para correr una simulación completa a partir de sus datos (o los de la NASA), ejecute el siguiente script (o archivo main-MiRoya-Dyn LAI XX.R):

```
rm(list = ls())
source("func-MiRoya-DynaCof.R")
#parameters
cormasInit <- "init1P_xx" #Name of the initialization method
for MiRoya
S0_sim <- "S0_xx_2013-2015.rda" #Name of the basic simulation
without rust
randomSeed <- NA #NA or a integer between 10.000 and 9.999.999

#@00000000@ XX @000000000000@ 
initAndRunOneYear(S0_sim)
# After this first year, run full simulation
runYears( S$Meteo$year[nrow(S$Meteo)] - S$Meteo$year[1])
```

Todos los datos de la simulación se almacenan en S2\$sim.

plotLAI(S, S2) :



Simulación interactiva

Es posible utilizar el modelo de forma interactiva, introduciendo acciones durante la simulación, como la aplicación de fungicida o fertilizante.

El principio es iniciar una simulación desde R y ejecutar un año sin interacción. A partir de este momento, es posible enviar diversas instrucciones para :

- simular 1 semana = runWeeks(1)
- simular 1 mes = runMonths(1)
- simular 1 año = runYears(1)
- fertilizar = fertilization()
- aumentar la calidad de las aplicaciones de fungicidas = capacitacionFungicidas()
- aplicar fungicidas sistémicos = aplicarFungicidaSistemica()
- aplicar fungicidas de protección = aplicarFungicidaProtector()

Las siguientes instrucciones muestran cómo utilizar esto con los datos climáticos de Aquiares (ver archivo main-MiRoya-Dyn LAI Aquiares.R).

```
rm(list = ls())
source("func-MiRoya-DynaCof.R")

#@@@@@@@ AQUIARES 2009-2016 @@@@@@@@ 
initFullSystem(S0_Name = "S0_Aquiares_2009-2016.rda", MiRoyaInit =
"init1P_Aquiares_2009_2016")
runFirstYear()
# /// Manipulation ///
runYears(1)
runMonths(1)
runWeeks(1)
fertilization()
capacitacionFungicidas() # increase Quality of fungicides by 10%
(initial Quality = 50%)
capacitacionCafetal() # do nothing
aplicarFungicidaSistemica()
aplicarFungicidaProtector()
usoFungicidaSistemica(1) # use systemic (1) or protective (0)
fungicides
aplicarFungicida(0.9) #apply fungicides (1 or 0) with a quality of
90%.
podaCafe_25() #pruning 25%
podaCafe_50() #pruning 50%
podaCafe(70) #pruning 70%, (pruning is performed at the next step)

# ****

```

Así, después de simular un primer año sin interacción, se puede ejecutar cada uno de los métodos de interacción en cualquier orden. En el siguiente gráfico, podemos seguir la evolución de la LAI que muestra la fuerte caída de la LAI tras una poda del 50%:



En realidad, las instrucciones de manipulación en R se traducen en funciones en MiRoya (ver archivo func-MiRoya-DynaCof.R). Así por ejemplo, la instrucción "fertilization()" corresponde a una llamada al método fertilization: num en MiRoya.

Configuración y ejecución de un análisis de sensibilidad

Para analizar los efectos de una variable, es aconsejable realizar una sucesión de simulaciones variando el valor de un parámetro en cada una de ellas. Esto se llama análisis de sensibilidad OAT (*One At a Time*, Uno por uno), es decir, variando el valor de un solo parámetro.

El siguiente script describe cómo ejecutar un plan de análisis en los datos climáticos de Aquiares variando el parámetro `probaInfeccionViento` de 0 a 100 (la probabilidad de alo-infección⁶; el valor estándar está fijado en 10%).

```

-- Do a simulation plan --
#-----
rm(list = ls())
source("func-MiRoya-DynaCof.R")
#parameters
nombreParametro <- "probaInfeccionViento" # Set NA for repeating
stochastic simulations"
shortNameParameter <- "pViento"
placeName <- "LaFe"
cormasInit <- "init1P_LaFe"
S0_sim <- "S0_sim_LaFe_2014-2020.rda" #Name of the basic simulation
without rust
randomSeed <- 3788179 #NA or a integer between 10.000 and 9.999.999

# Settings
S0_file <- paste(getwd(), "/outputs/", S0_sim, sep = "")
load(S0_file)
S0$Sim$Date <- S0$Meteo$Date
S=S0
resultsRoya <- initMatrix(S)
resultsLAI <- resultsRoya
resultsSitios <- resultsRoya
autoCafPorCm <- resultsRoya

# --- LOOP ---
numSimu <- 0
for (parametro in seq(from=0, to=1, by=0.01)){
  initFullSystem(S0_sim, cormasInit)
  runFirstYear()
  deactivateFertilization()
  if(!is.na(randomSeed)) { r <- setNumericAttributeValue("randomSeed:",
"MiRoya_Dynacof", randomSeed) }
  if(!is.na(nombreParametro)) {
    r <- setNumericAttributeValue((paste(nombreParametro, ":", sep = "")), "MiRoya_Dynacof", parametro)

    cat("***** Parametro ", nombreParametro, " = ", parametro, "***** \n")
    # run ONE SIMU
    runYears(S$Meteo$year[nrow(S$Meteo)] - S$Meteo$year[1])
}

```

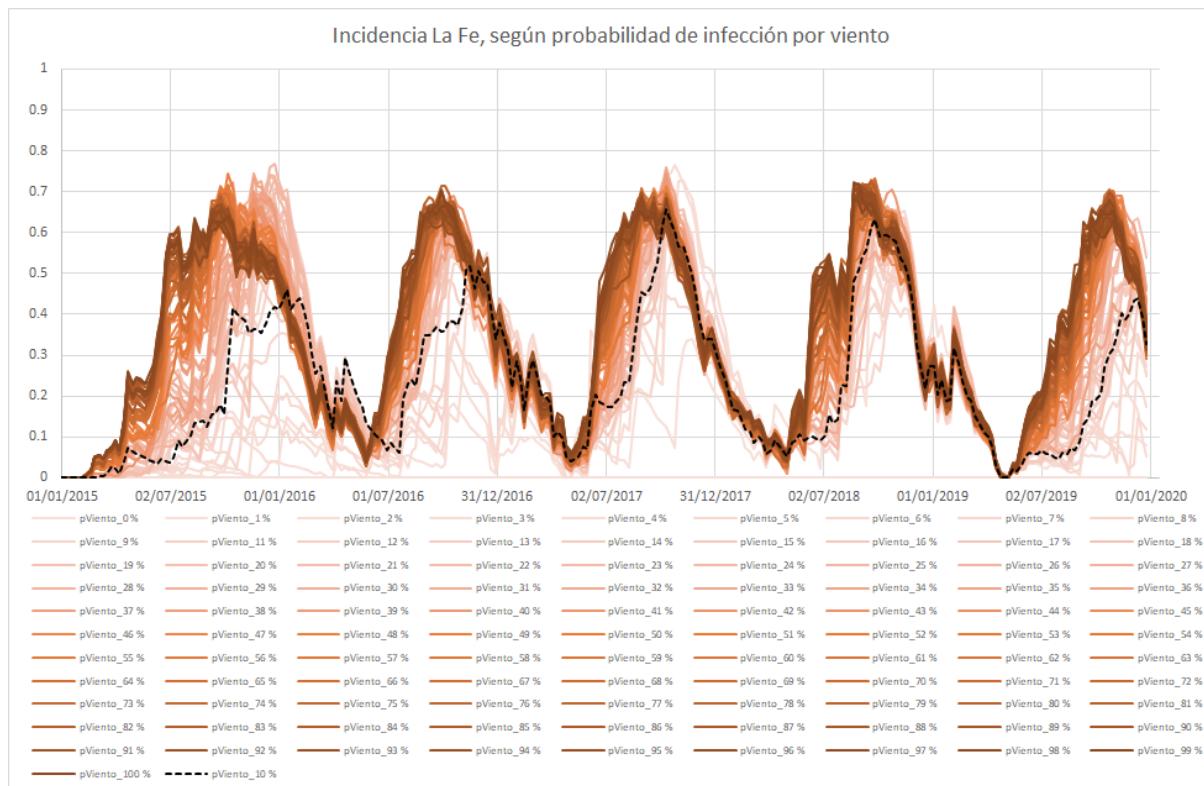
⁶ El método `#inocularEsporasPorViento` de MiRoya calcula el número de esporas procedentes del exterior de la parcela depositadas en las hojas: cada día, 10% (Hongo `probaInfeccionViento=0.1`) de probabilidad que 1% (Hongo `proporcionHojasViento=0.01`) de las hojas (aleatoriamente) reciben 5 esporas del exterior (Hongo `numDeposicionPorHoja=5`). Esto significa que en un día dado, o bien no hay hojas infectadas externamente (90% de probabilidad), o bien hay una chance de 1 en 10 de que el 1% de las hojas reciban esporas exógenas.

```

resultsRoya      <-      storeMatrix(resultsRoya,           "incidenciaRoya",
shortNameParameter, parametro)
resultsSitios    <-      storeMatrix(resultsSitios,       "porHoja_maxSitios",
shortNameParameter, parametro)
resultsLAI <-  storeMatrixLAI(resultsLAI, shortNameParameter, parametro)
} # --- End of the loop
write.csv(resultsRoya,file=paste(getwd(), "/outputs/incid", placeName, "_",
shortNameParameter, ".csv", sep = ""))
write.csv(resultsLAI,file=paste(getwd(), "/outputs/LAI-", placeName, "_",
shortNameParameter, ".csv", sep = ""))
write.csv(resultsSitios,file=paste(getwd(), "/outputs/Sitios-", placeName,
"_", shortNameParameter, ".csv", sep = ""))
#End of the simulation plan
#-----

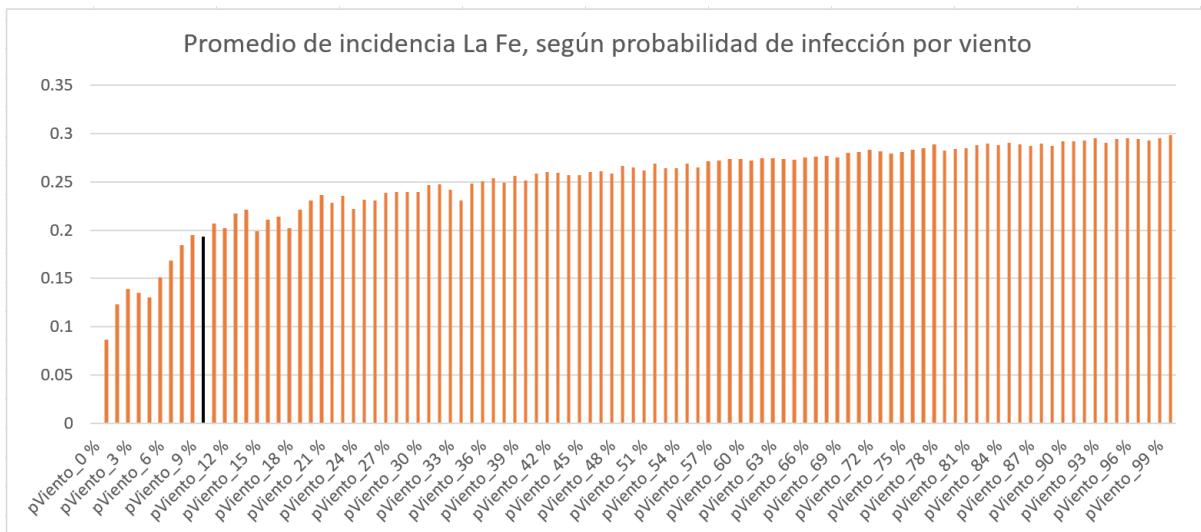
```

Los resultados (de incidencia, de LAI y de número de sitios por hojas) se almacenan en los archivos `incidLaFe_pViento.csv`, `LAI-LaFe_pViento.csv` y `Sitios-LaFe_pViento.csv`, en la carpeta `/outputs/`. Por ejemplo, el análisis anterior muestra los cambios en el indicador "incidencia" en función de pViento en el periodo 2015 - 2019:

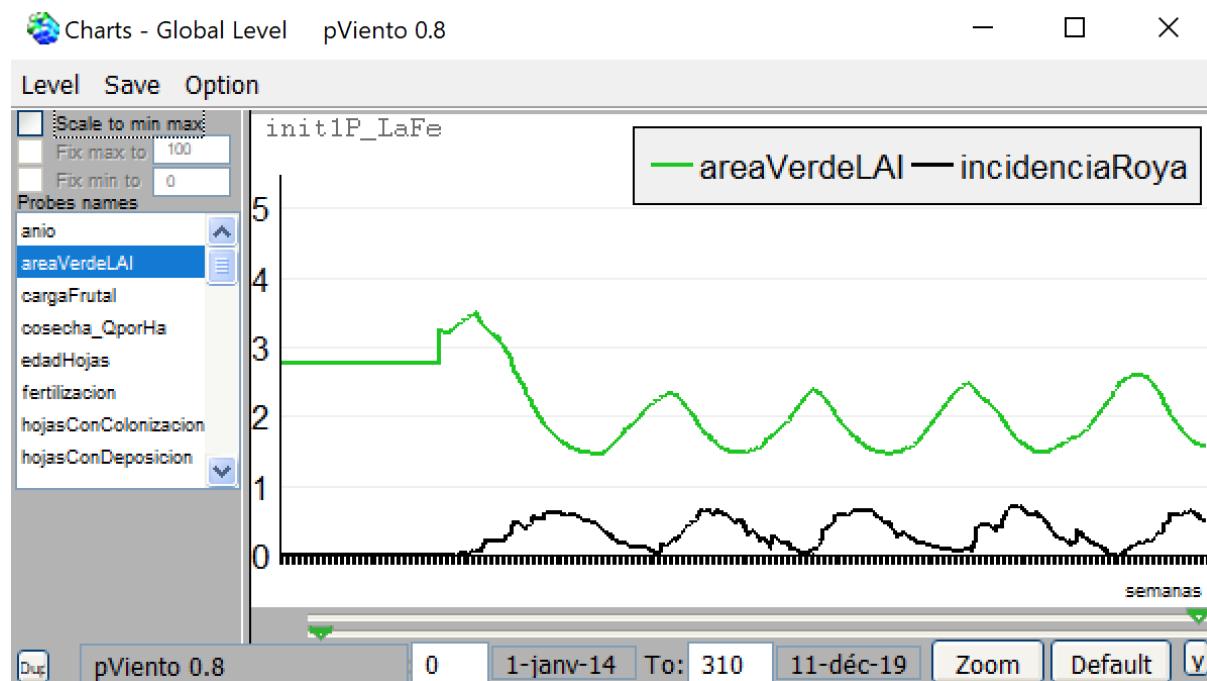


Evolución de la incidencia en función del parámetro probabilidad de aloinfección por el viento. La curva negra punteada corresponde al valor de referencia del modelo (10%).

Excepto en el caso de las probabilidades de infección muy bajas, se observa que, independientemente del valor de este parámetro, la dinámica de la incidencia sigue un patrón de fluctuaciones. El siguiente gráfico muestra el promedio de incidencia por cada simulación en función de la probabilidad de aloinfección:



También, al final de cada simulación, Cormas abre automáticamente una ventana de indicadores. Por ejemplo, al final de la simulación para el parámetro $pViento = 80\%$, se muestra la siguiente ventana:



Conclusión

MiRoya permite simular la dinámica de la roya y sus impactos en término de incidencia, caída de hojas y producción de café. Se trata de un modelo mecanístico para comprender y evaluar el funcionamiento de la roya del café y los parámetros que influyen en la propagación de las epidemias. Entonces, *MiRoya* es un modelo científico para probar hipótesis y explorar mediante simulación la influencia de diferentes parámetros en las posibles evoluciones del sistema café-roya. Las simulaciones informáticas ayudan a encontrar los factores que más influyen en este sistema y a anticipar los posibles efectos de una acción.

Destinado a los científicos, este modelo no es sencillo de instalar y utilizar. Para cualquier información adicional, puede ponerse en contacto con los autores.

Anexo

Parámetros y salidas de DynaCof

Lista de todas las variables de salidas. Instrucción = names(sim\$Sim)

```
> names(S$Sim)
[1] "Cycle"
[4] "LAI"
[7] "CM_FRoot"
[10] "CM_Fruit"
[13] "Rm"
[16] "SC"
"CM_Fruit_Cohort_remain"
[19] "CM_Fruit_Cohort"
"Alloc_Fruit_Cohort"
[22] "BudBreak_cohort"
[25] "Budinit"
[28] "Mprun_Shoot"
[31] "DM_RE"
[34] "DM_Fruit"
[37] "DM_Leaf"
[40] "NPP"
[43] "Rm_FRoot"
[46] "Mnat_FRoot"
[49] "Alloc_FRoot"
[52] "Rm_Leaf"
[55] "MnatALS_Leaf"
[58] "Rg_Leaf"
[61] "Supply_Leaf"
[64] "Rg_Fruit"
[67] "Alloc_Fruit"
[70] "Harvest_Maturity_Pot"
[73] "Rg_SCR"
[76] "lambdaSCRage"
[79] "Mnat_Shoot"
[82] "Alloc_Shoot"
[85] "Consumption_RE"
[88] "GPP"
[91] "Tcan_Diurnal_Cof_deg"
[94] "Rn_tot"
[97] "Rn_Coffee"
[100] "ETR"
[103] "RootWaterExtract_1"
[106] "IntercMax"
[109] "SoilWaterPot"
[112] "PAR_Soil"
[115] "TairCanopy"
[118] "Tleaf_Coffee"
[121] "H_tot"
[124] "Q_Soil"
[127] "LE_Plot"
[130] "EW_tot"
[133] "REW_2"
[136] "EW_1"
[139] "Drain_1"
[142] "TotSuperficialRunoff"
[145] "IntercRevapor"
[148] "W_2"
[151] "CanopyHumect"
"Plot_Age"
"LAIpolt"
"Harvest_Maturity"
"Demand_Fruit"
"CM_RE"
"Maturation_duration"
"NPP_Fruit_Cohort"
"Bud_available"
"pbreak"
"Mnat_SCR"
"DM_Fruit_Cohort"
"DM_Shoot"
"Temp_cor_Bud"
"Ra"
"Mortality_FRoot"
"Rg_FRoot"
"Supply_FRoot"
"Mortality_Leaf"
"MA_LoS"
"NPP_Leaf"
"Rm_Fruit"
"NPP_Fruit"
"Supply_Fruit"
"Mortality_SCR"
"NPP_SCR"
"Rm_Shoot"
"Rg_Shoot"
"Carbon_Lack_Mortality"
"K_Dir"
"lue"
"BudInitPeriod"
"Rn_Soil_SW"
"H_Coffee"
"RootWaterExtract_3"
"T_tot"
"AEu"
"PAR_Trans"
"APAR"
"WindSpeed_Coffee"
"Diff_T"
"LE_Tree"
"H_Soil"
"E_Soil"
"REW_3"
"EW_2"
"Drain_3"
"Infiltration"
"SuperficialRunoff"
"Throughfall"
"W_3"
"WSurfaceRes"
"Plot_Age_num"
"CM_Shoot"
"SM"
"CM_SCR"
"CM_Leaf"
"BudBreak"
"DegreeDays_Tcan"
"MAct_SCR"
"DM_SCR"
"DM_FRoot"
"Cbalance"
"Rg"
"Mprun_FRoot"
"NPP_FRoot"
"Demand_FRoot"
"Mprun_Leaf"
"MNat_Leaf"
"Alloc_Leaf"
"Harvest_Fruit"
"Overripened_Fruit"
"ratioNodestoLAI"
"Rm_SCR"
"Alloc_SCR"
"Mortality_Shoot"
"NPP_Shoot"
"Supply"
"K_Dif"
"NPP_RE"
"Date_harvest"
"LE_Coffee"
"SWD"
"RootWaterExtract_2"
"TCoffee"
"PSIL"
"PAR_Trans_Tree"
"APAR_Dif"
"TSoil"
"LE_tot"
"Rn_Soil"
"LE_Soil"
"REW_tot"
"EW_3"
"REW_1"
"Drain_2"
"InfiltCapa"
"ExcessRunoff"
"W_1"
"WTot"
"Height_Tree"
```

```
[154] "H_Tree"                      "Rn_Tree"                    "T_Tree"
[157] "APAR_Tree"                   "LAI_Tree"                   "TairCanopy_Tree"
[160] "Stocking_Tree"                "Height_Canopy"              "ALS"
[163] "Date"
```

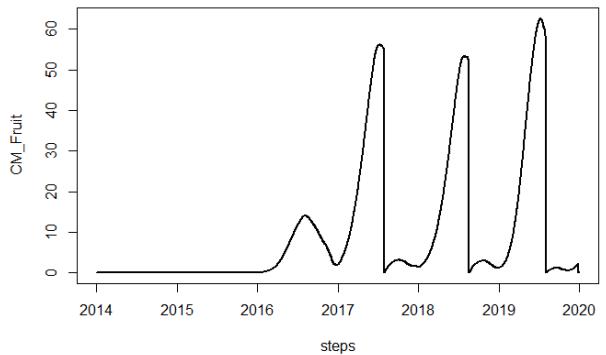
La descripción de las variables se encuentra en el archivo 2-main.R :

<https://github.com/VEZY/DynACof/blob/master/R/2-main.R>

```
CM_x          \tab gC m-2 d-1      \tab Organ C mass
NPP_x         \tab gC m-2 d-1      \tab Net primary productivity at organ
scale
DM_x          \tab gDM m-2 d-1     \tab Organ dry mass
```

Así, para trazar la variable CM_Fruit (C mass of Fruits, in gC m-2 d-1) :

```
var <- "CM_Fruit"
plot(S2[["Sim"]][[var]] ~ S2$Meteo$Date,
type="l", col="black", lwd=2, xlab="steps", ylab=var)
```



Dinamica del Cafeto en MiRoya sin Dynacof

En la versión actual, no se tiene en cuenta el crecimiento del cafeto. Sólo se consideran los cafetos adultos. Sólo se tiene en cuenta la generación (y caída) de hojas y frutos.

Diagrama de actividad del cafeto

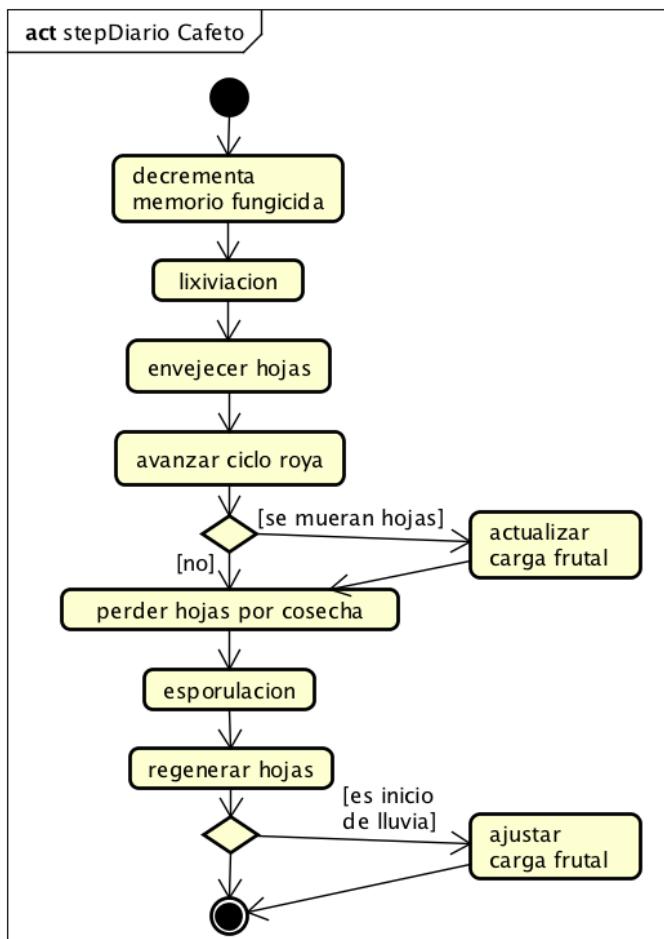


Fig. 12: Diagrama de actividad del cafeto

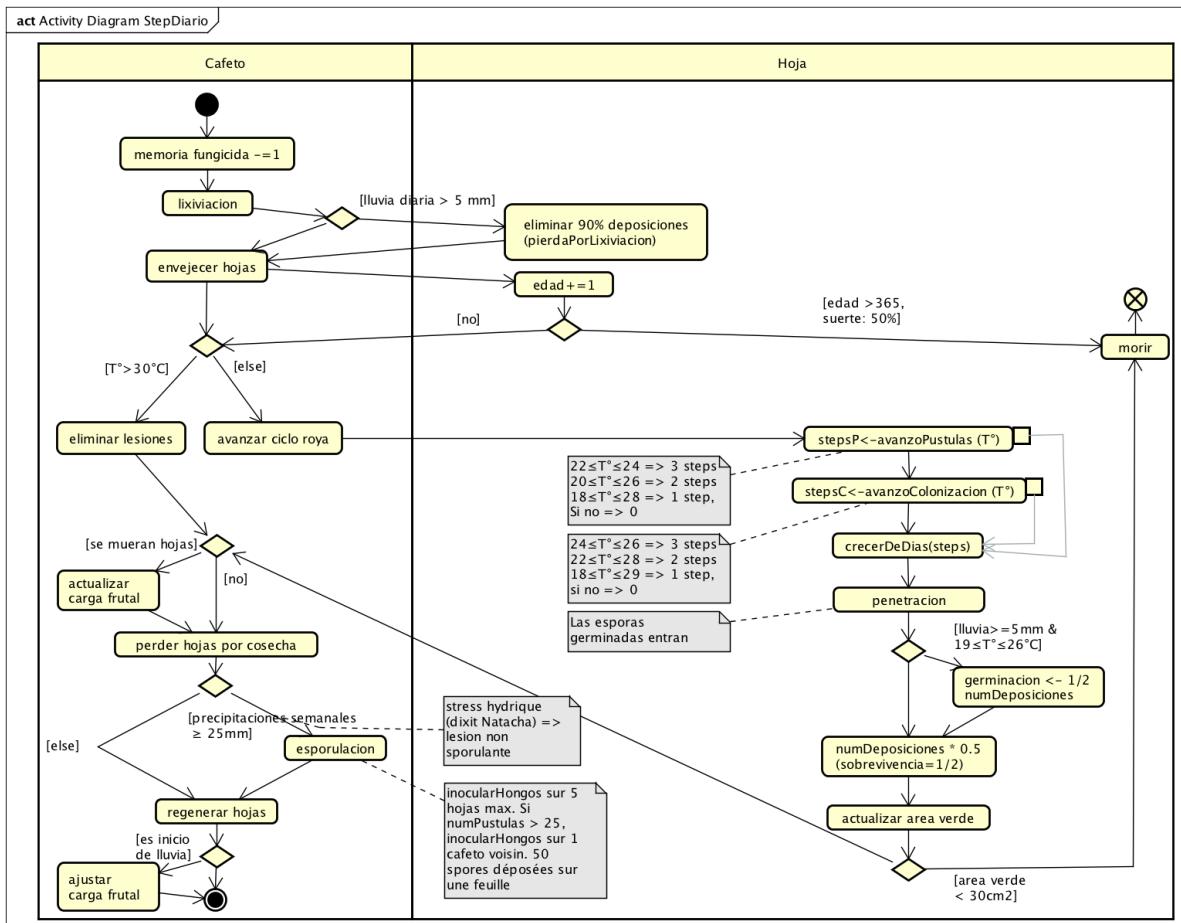


Fig. 13: Diagrama de actividad del cafeto y hoja (swimlanes)

powered by Astah

Fenología del cafeto

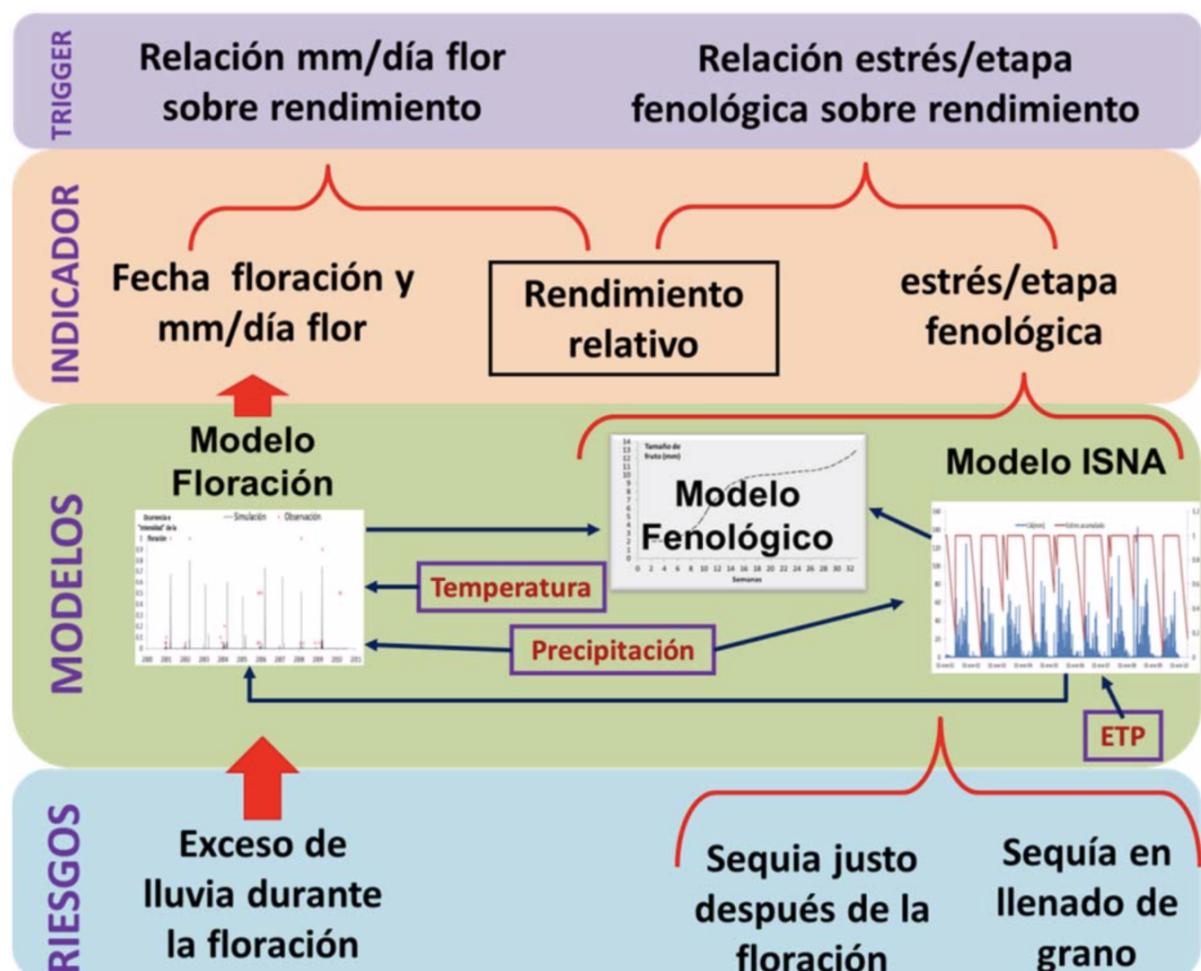


Fig 14: From: Leonel Lara, Bruno Rapidel, Dietmar Stoian, Jessenia Argüello, Thelma Gaitán, Claudia González, 2011. Estudio de factibilidad para la implementación de seguros basados en Índices climáticos en el cultivo de café en Honduras y Nicaragua. Technical Report · May 2011 DOI: 10.13140/RG.2.1.3976.9364

En el modelo, hay 4 estados:

- Floración: durante la primera semana de lluvia. La carga frutal maximal es calculada a este momento (a partir de la área verde del cafeto)
- Madurez: se acumula la temperatura (el $\Delta T^\circ = T^\circ - 10^\circ C$) hasta la madurez
- Cosecha = cuando $T\circ_{Acumulada} > umbral_{Madurez}$ ($3000^\circ C$)
- Desarrollo: de la cosecha hasta la floración.

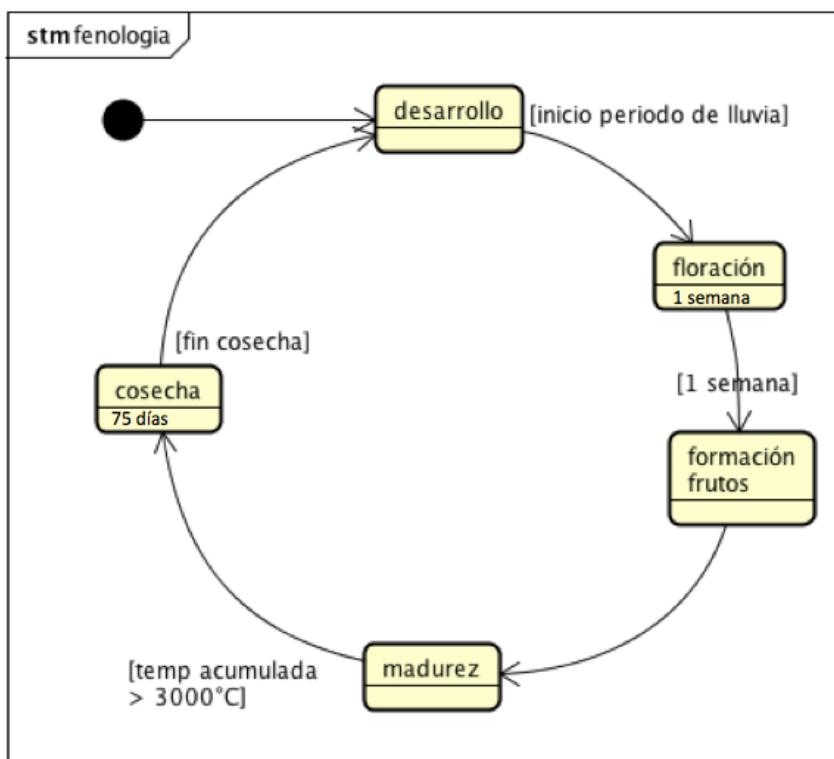


Fig. 15. Dinámica de la fenología

En la primera versión, el estado era un atributo del Cafeto. Pero hoy en día, el estado es un atributo de cada hoja porque dependiendo de las localidades y climas, un cafeto puede tener varios estados al mismo tiempo (frutos maduros y flores nuevas) ! Las flores aparecen en los nodos donde las hojas están en estado de desarrollo.

En el momento de la cosecha, sólo las hojas maduras con temperaturaAcumulada > 3000° se consideran "fructíferas", lo que da un porcentaje de la cosecha en relación con un óptimo (100 frutos). Ejemplo de El Salvador :

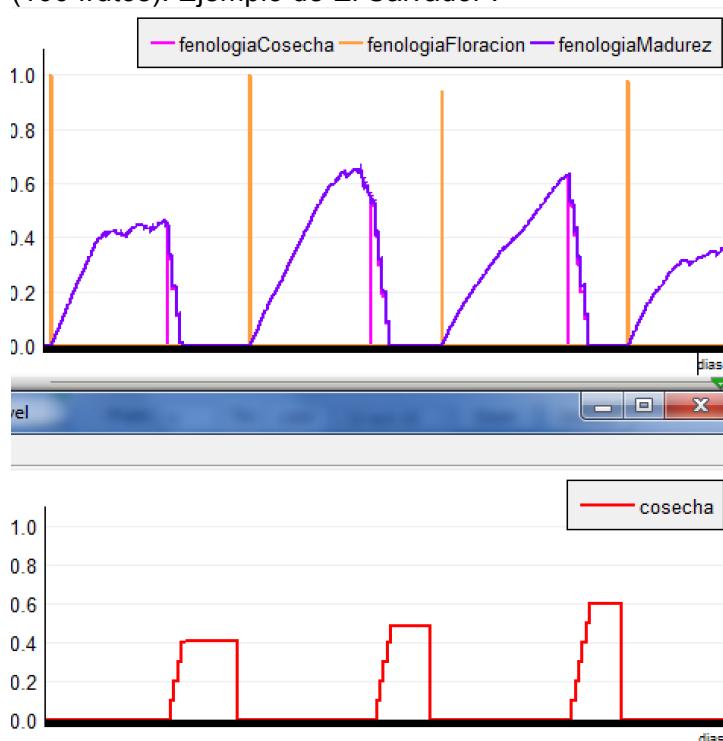


Fig. 16. Dinámica de la fenología y de la cosecha en 3 años de simulación

Regeneración de hojas

Cuando las precipitaciones semanales son mayores de 50 mm, entonces se generan 20 hojas nuevas.

Entre 20 y 50 mm/semana, se generan 10 hojas.
De lo contrario, sólo se crean 5 hojas por mes.

Caída de las hojas

Caída de hojas viejas

La esperanza de vida de una hoja es de 450 días. Más allá de eso, cada hoja tiene un 50% de posibilidades de morir.

Caída de hojas por cosecha

Cada árbol pierde hojas solamente durante el periodo de la cosecha (oct - dic): 10 hojas / mes.

Caída de hojas por roya

La Caída de las hojas sigue una simple ecuación estadística que depende de la incidencia de la roya:

tasa de defoliación diaria: $y = 0.0144 / \text{dia} / \text{incidencia}$;

Eso es el 10.08 % de las hojas infectadas que mueren cada semana.

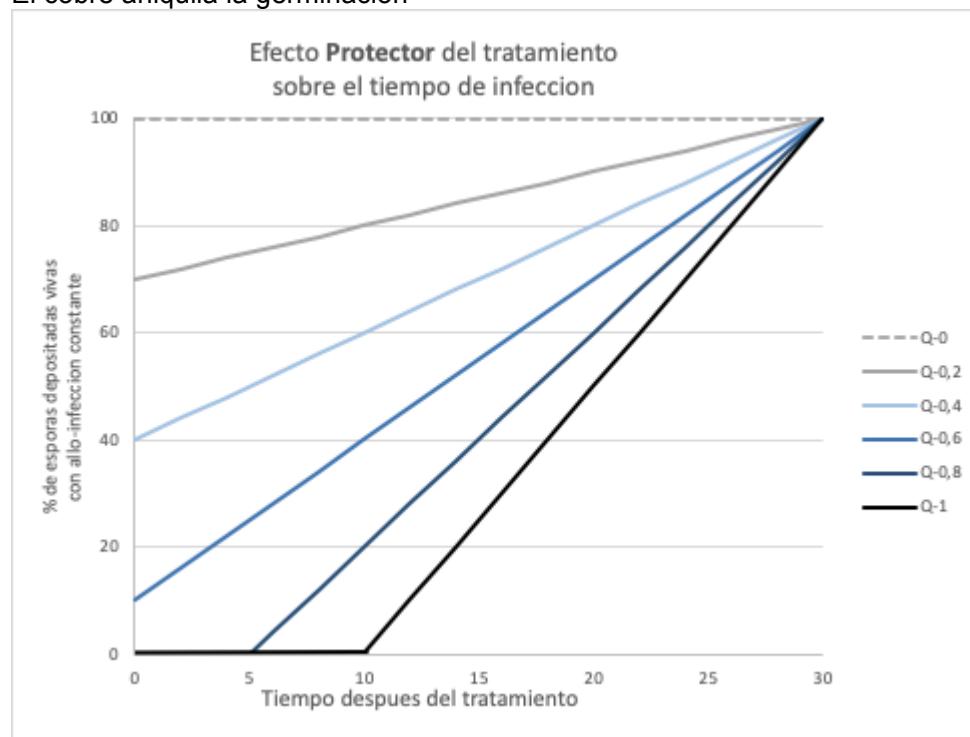
El cafeto pierde x hojas, las más afectadas (con lesiones esporulantes o no).

Acciones de la fungicida

Existen dos tipos de tratamiento fungicida: curativo (o sistémico) y protector.

Fungicida Protector

El cobre aniquila la germinación



Protector : $Y = A.T + B$.

$$\text{qd } T = 30, Y = 100\% \Rightarrow 100 = A.30 + B \Rightarrow A = (100-B) / 30$$

$$B=h(Q) : Q=0, B = 100 \text{ et } Q=1, B = -50$$

$$B = a.Q + b \Rightarrow B = a.Q + 100 \text{ et } B = a.1 + 100 = -50$$

$$a=-150$$

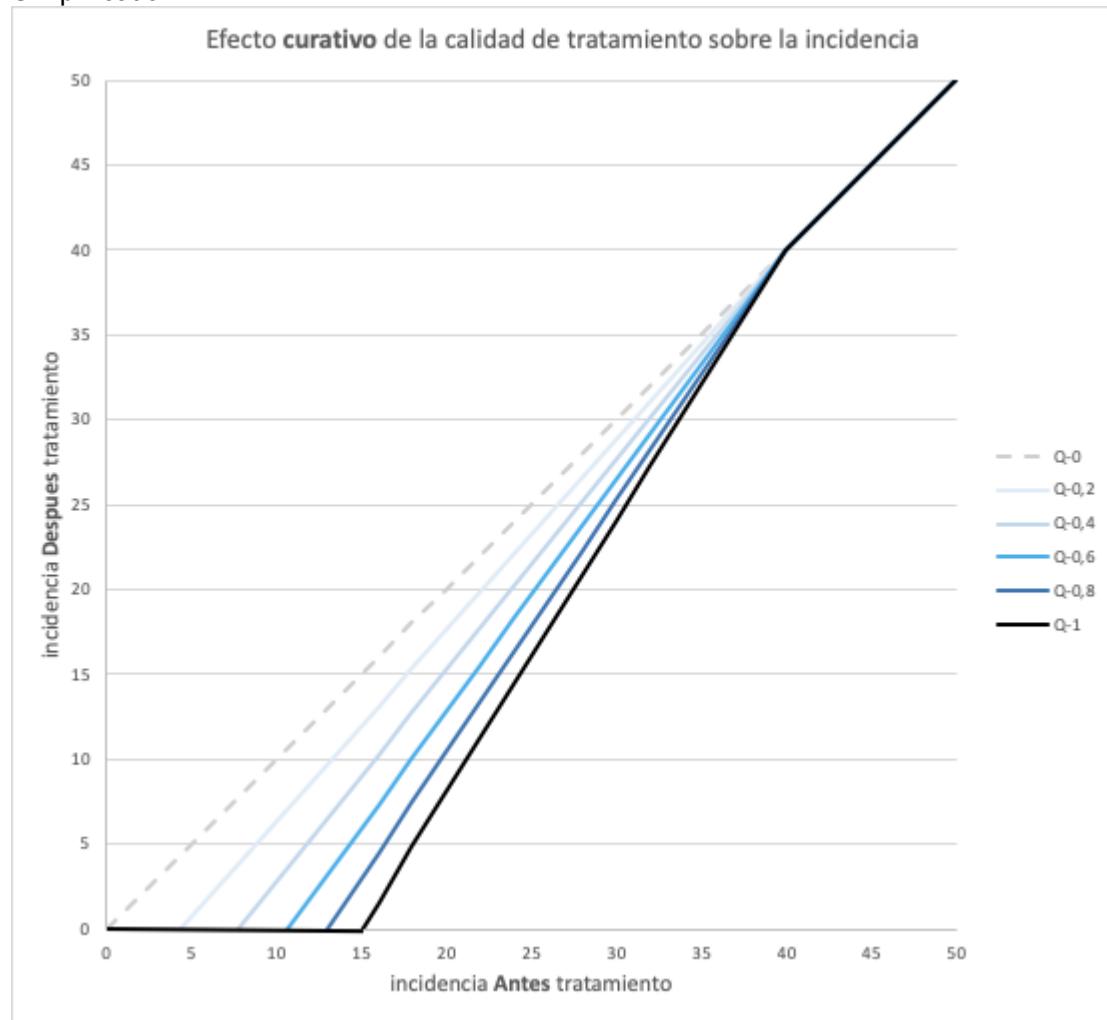
$$B = -150 . Q + 100$$

$$Y = (100 - B)/30 . T - 150.Q + 100 \Rightarrow Y = (100 + 150.Q - 100)/30 . T - 150.Q + 100$$

$$\Rightarrow Y = 5.Q . T - 150.Q + 100$$

Fungicida curativa, Triazol

Simplification :



Curativo : $Y = A.X + B$. $X=\text{incidencia_Antes}$, $Y=\text{incidencia_Despues}$.

$$\text{qd } x = 40\%, y=40\% \Rightarrow 40=A.40 + B \Rightarrow Y = (1-B/40) * x + B$$

$$\Rightarrow A = (40-B)/40 = 1 - B/40$$

$$B=g(Q) , B=a.Q + b \quad Q=0 \Rightarrow b = 0 \quad B = a.Q$$

$$Q=1 \Rightarrow b = -24 \quad B = -24.Q$$

$$Y = (1-B/40) * x + B \Rightarrow Y = (40 + 24Q)/40 * X - 24.Q$$

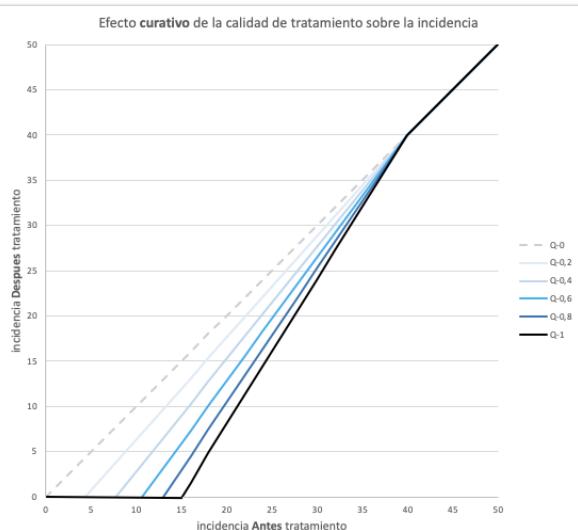
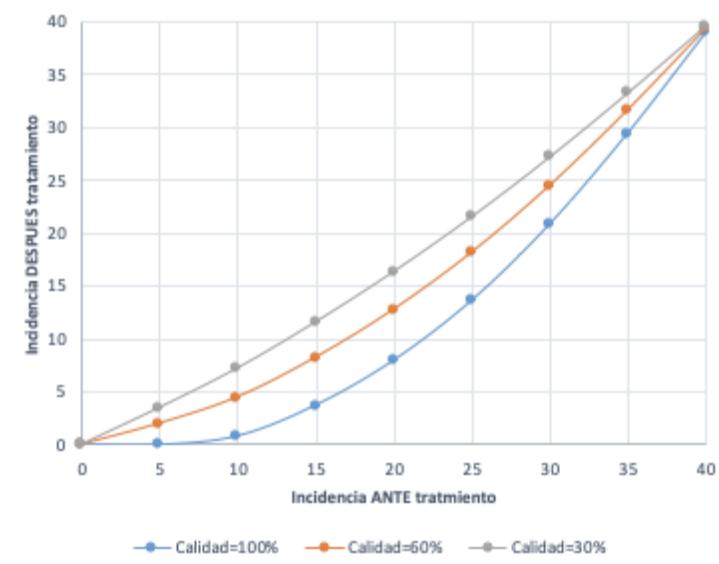
A esto, hay que añadir el efecto "protector" del fungicida curativo (o sistémico).

incidencia Despues = f(calidadTratamiento, incidenciaAntes), f = funcion cuadrática de incidenciaAntes.

$Q = \text{calidadTratamiento} = \text{valor entre } 0 \text{ y } 1$. IA = incidenciaAntes = %

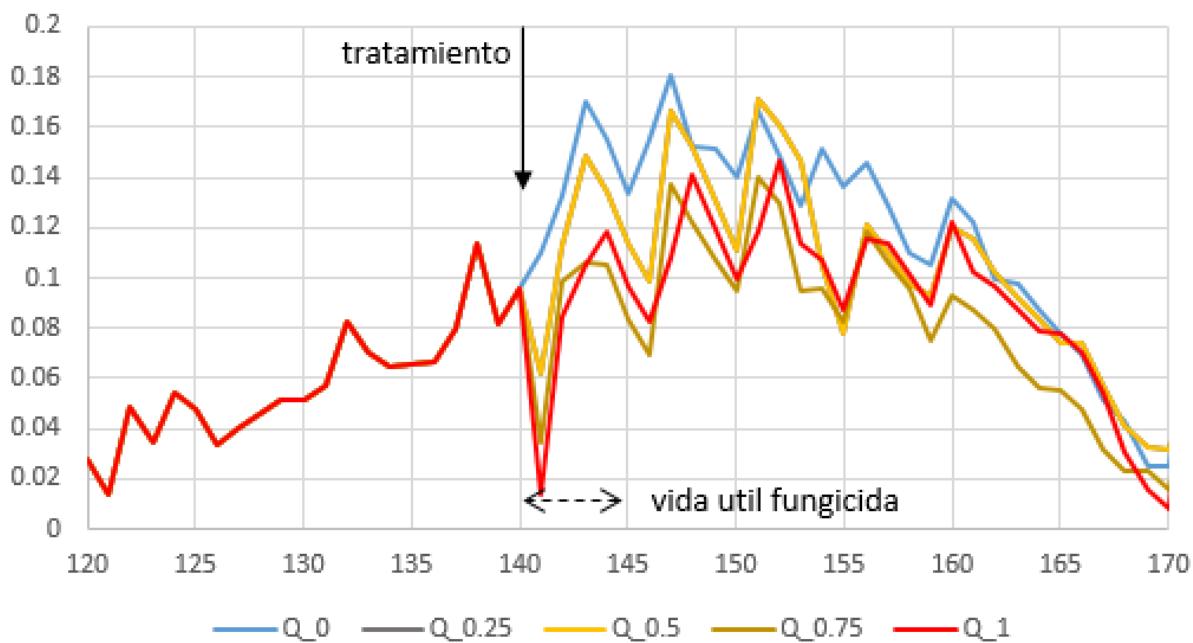
$$\text{incidencia} = 0.0289 * Q * IA^2 + (1 - (1.1759 * Q)) * IA - 0.0334 * IA$$

Efecto curativo de la calidad de tratamiento sobre la incidencia



Las lesiones que no fueron eliminadas por la primera acción del fungicida permanecen vivas pero no progresan.

Results : Capa= 1 traitement/an, Umbral = 10%

Incidencia = $F(Q)$ Deposicion por hoja = $f(Q)$ 