Recap computational structure of *possible*

$$DetSys : Set \to Set$$
$$DetSys\ X = X \to X$$

$$NonDetSys : Set \to Set$$
$$NonDetSys\ X = X \to List\ X$$

$$detFlow\ f\ zero = id$$
$$detFlow\ f\ (suc\ n) = detFlow\ f\ n \circ f$$

$$nonDetFlow\ f\ zero = \eta_{List}$$
$$nonDetFlow\ f\ (suc\ n) = f \mathbin{>\!\!=\!\!>_{List}} nonDetFlow\ f\ n$$

$$detTrj\ f\ zero\ x = x :: []$$
$$detTrj\ f\ (suc\ n)\ x = x :: detTrj\ f\ n\ (f\ x)$$

$$nonDetTrj\ f\ zero\ x = fmap_{List}\ (x ::\_)\ (\eta_{List}\ [])$$
$$nonDetTrj\ f\ (suc\ n)\ x = fmap_{List}\ (x ::\_)\ ((f \mathbin{>\!\!=\!\!>_{List}} (nonDetTrj\ f\ n))\ x)$$

Recap computational structure of *possible*

$$
\begin{aligned}
&\eta_{List} &&: \\
&\gg\!=\!>_{List} &&: \\
&fmap_{List} &&: (A \to B) \to List\ A \to List\ B \\
&\mu_{List} &&: List\ (List\ A) \to List\ A \\
&\gg\!=_{List} &&:
\end{aligned}
$$

$$
\begin{aligned}
&\eta_{List}\ x &&= [x] \\
&fmap_{List} &&= map \\
&nonDetToDet\ f\ xs &&= xs \gg\!=_{List} f
\end{aligned}
$$

$$
\begin{aligned}
&\forall\ (f : A \to B)\ (a : A) \to fmap_{List}\ f\ (\eta_{List}\ a) \equiv \eta_{List}\ (f\ a) \\
&\forall\ (as : List\ A) \to \mu_{List}\ (\eta_{List}\ as) \equiv as
\end{aligned}
$$

Monadic systems

Monadic systems

- Deterministic, non-deterministic, stochastic, etc. systems are instances of monadic systems

$$
\begin{array}{lll}
M & : & Set \to Set \\
fmap_M & : & \{A\ B : Set\} \to (A \to B) \to M\ A \to M\ B \\
\eta_M & : & \{A : Set\} \to A \to M\ A \\
\mu_M & : & \{A : Set\} \to M\ (M\ A) \to M\ A \\
\_{\gg=}_M\_ & : & \{B\ C : Set\} \to M\ B \to (B \to M\ C) \to M\ C \\
\_{>=>}_M\_ & : & \{A\ B\ C : Set\} \to (A \to M\ B) \to (B \to M\ C) \to (A \to M\ C)
\end{array}
$$

$$MonSys : Set \to Set$$

$$MonSys\ X = X \to M\ X$$

Monadic systems

- All results extend to monadic systems systems

$$
\begin{aligned}
&\mathit{monFlow} : \{X : \mathit{Set}\} \to \mathit{MonSys}\ X \to \mathbb{N} \to \mathit{MonSys}\ X\\
&\mathit{monFlow}\ f\ \ \mathit{zero} \quad = \eta_M\\
&\mathit{monFlow}\ f\ (\mathit{suc}\ n) = f \ggg\!\!=_M \mathit{monFlow}\ f\ n
\end{aligned}
$$

$$
\begin{aligned}
&\mathit{monTrj} : \{X : \mathit{Set}\} \to \mathit{MonSys}\ X \to (n : \mathbb{N}) \to X \to M\ (\mathit{Vec}\ X\ (\mathit{suc}\ n))\\
&\mathit{monTrj}\ f\ \mathit{zero} \quad\ x = \mathit{fmap}_M\ (x ::\_)\ (\eta_M\ [])\\
&\mathit{monTrj}\ f\ (\mathit{suc}\ n)\ x = \mathit{fmap}_M\ (x ::\_)\ (f\ x \gg\!= _M (\mathit{monTrj}\ f\ n))
\end{aligned}
$$

$$
\begin{aligned}
&\mathit{detToMon} : \{X : \mathit{Set}\} \to \mathit{DetSys}\ X \to \mathit{MonSys}\ X\\
&\mathit{detToMon}\ f = \eta_M \circ f
\end{aligned}
$$

Monadic systems

$$
\begin{aligned}
&monToDet \;:\; \{X \;:\; Set\} \;\to\; MonSys\; X \;\to\; DetSys\; (M\; X)\\
&monToDet\; f\; mx \;=\; mx \gg\!=_M f
\end{aligned}
$$

$$
\begin{aligned}
Det{\equiv}Mon \;:\; &\{X \;:\; Set\} \;\to\; (f \;:\; DetSys\; X) \;\to\; (n \;:\; \mathbb{N}) \;\to\; (x \;:\; X) \;\to\\
&\eta_M\; (detFlow\; f\; n\; x) \;\equiv\; monFlow\; (detToMon\; f)\; n\; x
\end{aligned}
$$

$$
\begin{aligned}
Mon{\equiv}Det \;:\; &\{X \;:\; Set\} \;\to\; (f \;:\; MonSys\; X) \;\to\; (n \;:\; \mathbb{N}) \;\to\; (mx \;:\; M\; X) \;\to\\
&monToDet\; (monFlow\; f\; n)\; mx \;\equiv\; detFlow\; (monToDet\; f)\; n\; mx
\end{aligned}
$$

- And more ...

The computational structure of *possible*: Monadic dynamical systems

The computational structure of *possible*: Monadic dynamical systems

- The bottom line is that, when the functor F is also a monad, $possible\ s\ :\ F\ Evolution$ can be defined in terms of computations like $monFlow$, $monTrj$ and their combinations

- Example 1: $Evolution\ =\ State,\ \ possible\ =\ monFlow\ next\ 5$

- Example 2: $Evolution\ =\ Vec\ State\ 5,\ \ possible\ =\ monTrj\ next\ 4$

- Example 3: $Evolution\ =\ State^2,\ \ possible\ =\ fmap_M\ (\lambda\ s'\ \to\ (s\ ,\ s'))\ (monFlow\ next\ 5\ s)$

Monadic systems (extra)

Monadic systems (extra)

- We have seen that the monadic operations fulfil certain equations, for example

$$\forall \ (f : A \to B)\,(a : A) \to fmap_{List}\ f\ (\eta_{List}\ a) \equiv \eta_{List}\ (f\ a)$$

- For arbitrary A, B, C : Set one has

$$\forall \ (f : A \to F\ B) \to \qquad (\_ \gg= f) \doteq \mu \circ fmap\ f$$
$$\forall \ (f : A \to F\ B) \to (g : B \to F\ C) \to f > => g \doteq \mu \circ fmap\ g \circ f$$
$$\mu \circ \eta \doteq id$$
$$\mu \circ fmap\ \eta \doteq id$$
$$\mu \circ \mu \doteq \mu \circ fmap\ \mu$$
$$\forall \ (f : A \to B) \to fmap\ f \circ \eta \doteq \eta \circ f$$
$$\forall \ (f : A \to B) \to fmap\ f \circ \mu \doteq \mu \circ fmap\ (fmap\ f)$$

- In this specification, $f \doteq g$ means that f is extensionally equal to g:

$$f \doteq g = (x : dom\ f) \to f\ x \equiv g\ x$$

Monadic systems (extra)

- Monadic laws are best understood diagrammatically

$$
\begin{tikzcd}
F\ C & M\ (M\ C) \arrow[l, "\mu_M"'] & M\ B \arrow[l, "fmap_M\ g"'] & A \arrow[l, "f"'] \arrow[lll, bend right, "f >=>_M g"']
\end{tikzcd}
$$

$$
\begin{tikzcd}
M\,X \arrow[r, "\eta_M"] \arrow[dr, "id"'] & M\,(M\,X) \arrow[d, "\mu_M"] & M\,X \arrow[l, "fmap_M\ \eta_M"'] \arrow[dl, "id"] \\
 & M\,X &
\end{tikzcd}
$$

$$
\begin{tikzcd}
M\,(M\,(M\,X)) \arrow[r, "fmap_M\ \mu_M"] \arrow[d, "\mu_M"'] & M\,(M\,X) \arrow[d, "\mu_M"] \\
M\,(M\,X) \arrow[r, "\mu_M"'] & M\,X
\end{tikzcd}
$$

$$
\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "\eta_M"'] & Y \arrow[d, "\eta_M"] \\
M\,X \arrow[r, "fmap_M\ f"'] & M\,Y
\end{tikzcd}
$$

$$
\begin{tikzcd}
M\,(M\,X) \arrow[r, "fmap_M\ (fmap_M\ f)"] \arrow[d, "\mu_M"'] & M\,(M\,Y) \arrow[d, "\mu_M"] \\
M\,X \arrow[r, "fmap_M\ f"'] & M\ Y
\end{tikzcd}
$$

Monadic systems (extra)

Exercise 4.14

Postulate the monadic laws in Agda.

Exercise 4.15

Using the postulated monadic laws, prove $Det{\equiv}Mon$. (It should be very similar to the earlier proof of $Det{\equiv}NonDet$, but now for an arbitrary monad.)