Resumen del artículo:

“Just do It: Simple Monadic Equational Reasoning”, Jeremy Gibbons and Ralf Hinze, Department of Computer Science, University of Oxford

En este artículo se presenta un estudio acerca de los efectos través del uso de monádicos en la programación, en donde se hace énfasis en la programación funcional.

Los lenguajes de programación funcional resultan ser buenos para el razonamiento por ecuaciones. En la actualidad, pocos trabajos se han realizado para estudiar la combinación del razonamiento por ecuaciones y la programación con el fin de analizar los efectos.

En este trabajo se presenta una simulación al realizar computación como una función matemática pura, en donde se conduce el razonamiento como un valor puro. También se muestra un razonamiento mediante ecuaciones a través de programas monádicos.

La idea principal es explotar las principales propiedades de las operaciones algebraicas de una interfaz monádica y las operaciones particulares usadas en una clase de efectos (operaciones como put, get, …).

Como aportación adicional, se presenta un razonamiento en programas que explotan no determinismo y selección probabilística.

Se aborda el problema de Torres de Hanoi mediante monádicos de la siguiente manera:

*hanoi :: MonadCount m => Int -> m ()*

*hanoi 0 = skip*

*hanoi (n + 1) = hanoi n >> tick >> hanoi n*

El no determinismo es tratado mediante una función de selección y otra que representa el fallo de un resultado. De igual forma se presentan funciones que tratan permutaciones y la computación probabilística es tratada mediante la distribución probabilística de un monádico.

Como conclusión se presentó un enfoque de programación funcional, en donde que se tratan clases de efectos como datos abstractos mediante operaciones algebraicas y ecuaciones.