Supplementary information

*De-novo modeling of additional subdomains*

D3a, D3b and D3c were modeled *de-novo* using MC-Sym. The most conserved element of D3 among group II introns is the basal stem (1,2), which was already modeled by homology with the OiIIC structure. In addition to the basal stem, D3 of the ai5yIIB intron extends into three additional stems, forming a four way junction. Among these stems, D3a has been shown to form a long-range interaction (µ-µ')(3) with D5. From the homology model, we noticed that the basal stem of D3 and D5 are almost perpendicular to each other. Specifically, G844 which is part of µ-µ' interaction is almost perpendicular to basal stem of D3. Hence, if the stem-loop of D3a interacts with G844, then D3a and the basal stem of D3 should be next to each other (mostly as parallel helices). We used this criterion to filter out various alternative conformations of D3 predicted by MC-Sym. Remaining conformations were docked on to the core model using the basal stem of D3 as the anchor, and the conformation with minimum distance between D5 and D3a was selected.

D1c2 and D1d2a are additional insertions involved in a long-range kissing loop interaction (β-β') found within more derived group II introns. The D1c2 is an insertion within the D1c1 helix, resulting in formation of a three-way junction, two helices of which are already present in the homology model. Again, we employed MC-Sym to predict various conformations of D1c2 and we docked the resulting structures onto the core model. A variety of D1d2a conformations were also predicted with MC-Sym and docked onto same model in parallel. We visually screened potential solutions and selected the final conformation in which the loops involved in β-β' are close to each other. Finally, the kissing loop was manually modeled as a helix.
**Figure S1:** SHAPE analysis of WT vs Mutant ai5γ. *(A)* SHAPE reactivities for WT D135 at 0 mM Mg$^{2+}$ (not folded, β-β’ interaction not formed) and 100 mM Mg$^{2+}$ (folded, β-β’ interactions formed). At 0 mM Mg$^{2+}$, nucleotides involved in β-β’ are highly reactive, as the kissing-loop interaction is not formed. *(B)* In the β-β’ mutant even at 100 mM Mg$^{2+}$ nucleotides (#145-149) involved in the β-β’ kissing-loop interaction showed a significant increase in SHAPE reactivities, this was expected as the other half of this kissing loop pair was replaced with a tetraloop. For the remaining regions of the mutant there were no significant changes in SHAPE reactivity when compared to WT, suggesting that the secondary structure has not been affected by the mutations.
Figure S2: Validation of D3. D3 is shown as a transparent surface representation, colored in yellow. Shown in red are the regions that showed loss in hydroxyl protection in the D15 construct (D3 deleted) but which are protected in D135 construct(4). Clearly the majority of these regions are located beneath D3 in the model, suggesting they are more exposed upon deletion of D3 in the D15 construct, thereby validating the location of D3. There are two regions that are far from D3 but less protected in D15 construct (nucleotides 142-146, 355-359). However, these nucleotides are poorly protected in all deletion other constructs studied to date (D1, D15, D135ζ')(4). Therefore loss of protection in these regions can be attributed to overall loosening of the D1 scaffold.
Figure S3: Modeling the η-η' interaction between D2 and D6. (A) D2 is colored as shown in the legend, the ‘active’ conformation of D6 is colored in green and the ‘silent’ conformation is colored in magenta and the rest of the intron is colored in gray. As discussed in the results section the toggling of D6 between ‘active’ (green) and ‘silent’ (magenta) conformations involves a simple rotation rather larger translational movement. We were able to model D2 such that slight conformational change in D6 is sufficient to support formation of the η-η' interaction. (B) Stereo view of the D2 model. D2 was modeled such that basal stem is coaxial with D2a and perpendicular to the stem D2b. The bend at this 4-way junction places D2b close to D6 and thereby facilitates formation of the η-η' interaction. The topology of the D2 4-way
junction is common and similar to the junction families previously described by Schlick as cX or cL (5). The entire secondary structure of D2 remains intact in both states, although the A463-U431 base pair was slightly opened to accommodate modeling of the η-η’ interaction. Nucleotides from #470-530 of D2 are not shown in both panels for clarity.
Figure S4: (A) Stereo view depicting the structural architecture of ai5yIIB. All domains and subdomains are color-coded same as the Figure 4 in the manuscript. (B) 180° rotated view of the image shown in (A).
**Table S1**: List of computational tools employed in each step of modeling and their role in the process.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Role</th>
<th>Additional details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1: Homology modeling</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BioEdit (6)</td>
<td>To perform sequence alignment between two RNAs manually.</td>
<td></td>
</tr>
<tr>
<td>ModeRNA (7)</td>
<td>To build a homology model from a given template structure and sequence alignment.</td>
<td></td>
</tr>
<tr>
<td>MC-Sym (8)</td>
<td>To rebuild helices as idealized A-form structures.</td>
<td>This step can be skipped if helices in target and template structure are of the same length.</td>
</tr>
<tr>
<td>PyMol (Schrödinger, LLC) and DS visualizer (Accelrys Software Inc)</td>
<td>To dock the helices rebuilt using MC-Sym</td>
<td>PyMol is used to align the helices and DS visualizer is employed to manually move the helices if required.</td>
</tr>
<tr>
<td>RCrane (9)</td>
<td>To refine the backbone after docking the helices</td>
<td>The process of replacing helices may introduce breaks in the backbone in such cases RCrane is employed to rebuild the backbone without disturbing the base-pairs.</td>
</tr>
<tr>
<td><strong>Step 2: De-novo modeling</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MC-Sym (8)</td>
<td>To model the missing domain <em>de-novo</em></td>
<td>In the input for MC-Sym we included a region that was already part of the homology model, and which served as an anchor for later docking steps. (see below for an example of MC-Sym Script)</td>
</tr>
<tr>
<td>PyMol (Schrödinger, LLC) and DS visualizer (Accelrys Software Inc)</td>
<td>To dock domains built by MC-Sym onto the homology model</td>
<td>PyMol was employed to align the anchor regions in the <em>de-novo</em> model onto the corresponding region in the homology model. DS visualizer is employed to manually move the helices if required.</td>
</tr>
<tr>
<td><strong>Step 3: Model Refinement</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCrane (9)</td>
<td>To refine the backbone after docking the missing domains.</td>
<td></td>
</tr>
<tr>
<td>AMBER (10)</td>
<td>To minimize the structure and remove steric overlaps.</td>
<td></td>
</tr>
<tr>
<td>MolProbity (11)</td>
<td>To assess the quality of the resulting model.</td>
<td></td>
</tr>
</tbody>
</table>
Example of MC-Sym script

// ==================== Sequence =================

sequence( r A1
GAGCCGUAUGCAUGAAAGUCGCACGUACGGUUCUUACCGGGAAAACUUGUAAAGGUCUACGUACCUAUCGG )

((((((((((((((....)))))..)))))))))...(((((((....((((....)))))))...))))

BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB

// 1234567890123456789012345678901234567890123456789012345678901234567890
// 1 2 3 4 5 6 7

// ==================== Library ===============

//-------- Fragment A ------
// stem 1 --> 14, 19 --> 34

ncm_01 = library(
    pdb( "MCSYM-DB/2_2/GAUC/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A1:A2, A33:A34
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_02 = library(
    pdb( "MCSYM-DB/2_2/AGUU/*_1.pdb.gz" )
    #1:#2, #3:#4 <- A2:A3, A32:A33
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_03 = library(
    pdb( "MCSYM-DB/2_2/GCGU/*_1.pdb.gz" )
    #1:#2, #3:#4 <- A3:A4, A31:A32
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_04 = library(
    pdb( "MCSYM-DB/2_2/CCGG/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A4:A5, A30:A31
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_05 = library(
    pdb( "MCSYM-DB/2_2/CGCG/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A5:A6, A29:A30
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_06 = library(
    pdb( "MCSYM-DB/2_2/GUAC/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A6:A7, A28:A29
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_07 = library(
pdb( "MCSYM-DB/2_2/UAUA/*R20*.pdb.gz" )
#1:#2, #3:#4 < A7:A8, A27:A28
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_08 = library(
  pdb( "MCSYM-DB/2_2/AUGU/*_1.pdb.gz" )
  #1:#2, #3:#4 < A8:A9, A26:A27
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_09 = library(
  pdb( "MCSYM-DB/2_4/UGCACG/*.pdb.gz" )
  #1:#2, #3:#6 < A9:A10, A26:A27
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_10 = library(
  pdb( "MCSYM-DB/2_2/GCGC/*R20*.pdb.gz" )
  #1:#2, #3:#4 < A10:A11, A22:A23
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_11 = library(
  pdb( "MCSYM-DB/2_2/CGCG/*R20*.pdb.gz" )
  #1:#2, #3:#4 < A11:A12, A21:A22
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_12 = library(
  pdb( "MCSYM-DB/2_2/GAUC/*R20*.pdb.gz" )
  #1:#2, #3:#4 < A12:A13, A20:A21
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_13 = library(
  pdb( "MCSYM-DB/2_2/AUGU/*_1.pdb.gz" )
  #1:#2, #3:#4 < A13:A14, A19:A20
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

ncm_14 = library(
  pdb( "MCSYM-DB/6/UGAAAG/*.pdb.gz" )
  #1:#6 < A14:A19
  rmsd( 0.5 sidechain && !( pse || lp || hydrogen ) ) )

//------ Fragment B ------
// stem 38 -> 51, 56 -> 70

lnk_01 = library(
  pdb( "MCSYM-DB/ss2/CU/*.pdb.gz" )
  #1:#2 < A34:A35
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )

lnk_02 = library(
  pdb( "MCSYM-DB/ss2/UU/*.pdb.gz" )
  #1:#2 < A35:A36
  rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) ) )
lnk_03 = library(
    pdb( "MCSYM-DB/ss2/UA/*.pdb.gz" )
    #1:#2 <- A36:A37
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

lnk_04 = library(
    pdb( "MCSYM-DB/ss2/AC/*.pdb.gz" )
    #1:#2 <- A37:A38
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_15 = library(
    pdb( "MCSYM-DB/2_2/CCGG/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A38:A39, A69:A70
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_16 = library(
    pdb( "MCSYM-DB/2_2/CGCG/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A39:A40, A68:A69
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_17 = library(
    pdb( "MCSYM-DB/2_2/GGUC/*_1.pdb.gz" )
    #1:#2, #3:#4 <- A40:A41, A67:A68
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_18 = library(
    pdb( "MCSYM-DB/2_3/GGUAU/*.pdb.gz" )
    #1:#2, #3:#5 <- A41:A42, A65:A67
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_19 = library(
    pdb( "MCSYM-DB/2_2/GGCU/*_1.pdb.gz" )
    #1:#2, #3:#4 <- A42:A43, A64:A65
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

ncm_20 = library(
    pdb( "MCSYM-DB/2_2/GGCC/*R20*.pdb.gz" )
    #1:#2, #3:#4 <- A43:A44, A63:A64
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

lnk_05 = library(
    pdb( "MCSYM-DB/ss2/GA/*.pdb.gz" )
    #1:#2 <- A44:A45
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

lnk_06 = library(
    pdb( "MCSYM-DB/ss2/AA/*.pdb.gz" )
    #1:#2 <- A45:A46
    rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )
)

lnk_07 = library(
    pdb( "MCSYM-DB/ss2/AA/*.pdb.gz" )
)
#1:#2 <- A46:A47
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

lnk_08 = library(
pdb( "MCSYM-DB/ss2/AA/*.pdb.gz" )
#1:#2 <- A47:A48
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

cmc_21 = library(
pdb( "MCSYM-DB/2_2/ACGU/*R20*.pdb.gz" )
#1:#2, #3:#4 <- A48:A49, A58:A59
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

cmc_22 = library(
pdb( "MCSYM-DB/2_2/CUGG/*_1.pdb.gz" )
#1:#2, #3:#4 <- A49:A50, A57:A58
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

cmc_23 = library(
pdb( "MCSYM-DB/2_2/UUAG/*_1.pdb.gz" )
#1:#2, #3:#4 <- A50:A51, A56:A57
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

cmc_24 = library(
pdb( "MCSYM-DB/6/UGUAAA/*.pdb.gz" )
#1:#6 <- A51:A56
rmsd( 0.5 sidechain && !( pse || lp || hydrogen ) )

//----- Threading -----

lnk_09 = library(
pdb( "MCSYM-DB/ss2/AC/*.pdb.gz" )
#1:#2 <- A62:A63
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

lnk_10 = library(
pdb( "MCSYM-DB/ss2/UA/*.pdb.gz" )
#1:#2 <- A61:A62
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

lnk_11 = library(
pdb( "MCSYM-DB/ss2/CU/*.pdb.gz" )
#1:#2 <- A60:A61
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

lnk_12 = library(
pdb( "MCSYM-DB/ss2/AC/*.pdb.gz" )
#1:#2 <- A37:A38
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

lnk_13 = library(
pdb( "MCSYM-DB/ss2/UA/*.pdb.gz" )
# A36:A37
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

# A35:A36
rmsd( 0.1 sidechain && !( pse || lp || hydrogen ) )

// ===================== Backtrack =====================
// assemble the whole structure:

structure = backtrack

//------- Fragment A ------
// stem 1 -> 14, 19 -> 34
ncm_01
merge( ncm_02 1.5 )
merge( ncm_03 1.5 )
merge( ncm_04 1.5 )
merge( ncm_05 1.5 )
merge( ncm_06 1.5 )
merge( ncm_07 1.5 )
merge( ncm_08 1.5 )
merge( ncm_09 1.5 )
merge( ncm_10 1.5 )
merge( ncm_11 1.5 )
merge( ncm_12 1.5 )
merge( ncm_13 1.5 )
merge( ncm_14 1.5 )

//------- Fragment B ------
// stem 38 -> 51, 56 -> 70
// placed with respect to fragment A via 34 -> 38
merge( lnk_01 1.5 )
merge( lnk_02 1.5 )
merge( lnk_03 1.5 )
merge( lnk_04 1.5 )
merge( ncm_15 1.5 )
merge( ncm_16 1.5 )
merge( ncm_17 1.5 )
merge( ncm_18 1.5 )
merge( ncm_19 1.5 )
merge( ncm_20 1.5 )
merge( lnk_05 1.5 )
merge( lnk_06 1.5 )
merge( lnk_07 1.5 )
merge( lnk_08 1.5 )
merge( ncm_21 1.5 )
merge( ncm_22 1.5 )
merge( ncm_23 1.5 )
merge( ncm_24 1.5 )

//------ Threading ------
// to complete the rest of the molecule
// NB these could be added later...
merge( lnk_09 1.5 )
merge( lnk_10 1.5 )
merge( lnk_11 1.5 )
merge( lnk_12 1.5 )
merge( lnk_13 1.5 )
merge( lnk_14 1.5 )

// ================ Distance Restraints ================
// these distance restraints will ease the construction

distance( A59:C1' A63:C1' 0.0 22.4 )
distance( A59:C1' A61:C1' 0.0 13.6 )
distance( A59:C1' A62:C1' 0.0 18.0 )
distance( A61:C1' A63:C1' 0.0 13.6 )
distance( A60:C1' A63:C1' 0.0 18.0 )

distance( A34:C1' A38:C1' 0.0 22.4 )
distance( A34:C1' A36:C1' 0.0 13.6 )
distance( A34:C1' A37:C1' 0.0 18.0 )
distance( A36:C1' A38:C1' 0.0 13.6 )
distance( A35:C1' A38:C1' 0.0 18.0 )

// ============ Backtrack Restraints ============

clash
(
  structure
    1.5 !( pse || lp || hydrogen )
)

backtrack_rst
(
  structure
    width_limit = 25%,
    height_limit = 33%,
    method = probabilistic
)

// ============== Ribose Restraints ==============
ribose rst
{
  structure
  method    = ccm,
  pucker    = C3p_endo,
  threshold = 2.0
}

// =================== Exploration Initialization =========

explore
{
  structure
  option(
    model_limit = 1000,
    time_limit = 666d,
    seed        = 3210
  )
  rmsd( 3.0 sidechain && !( pse || lp || hydrogen ) )
  pdb( "d5d6" zipped )
}


