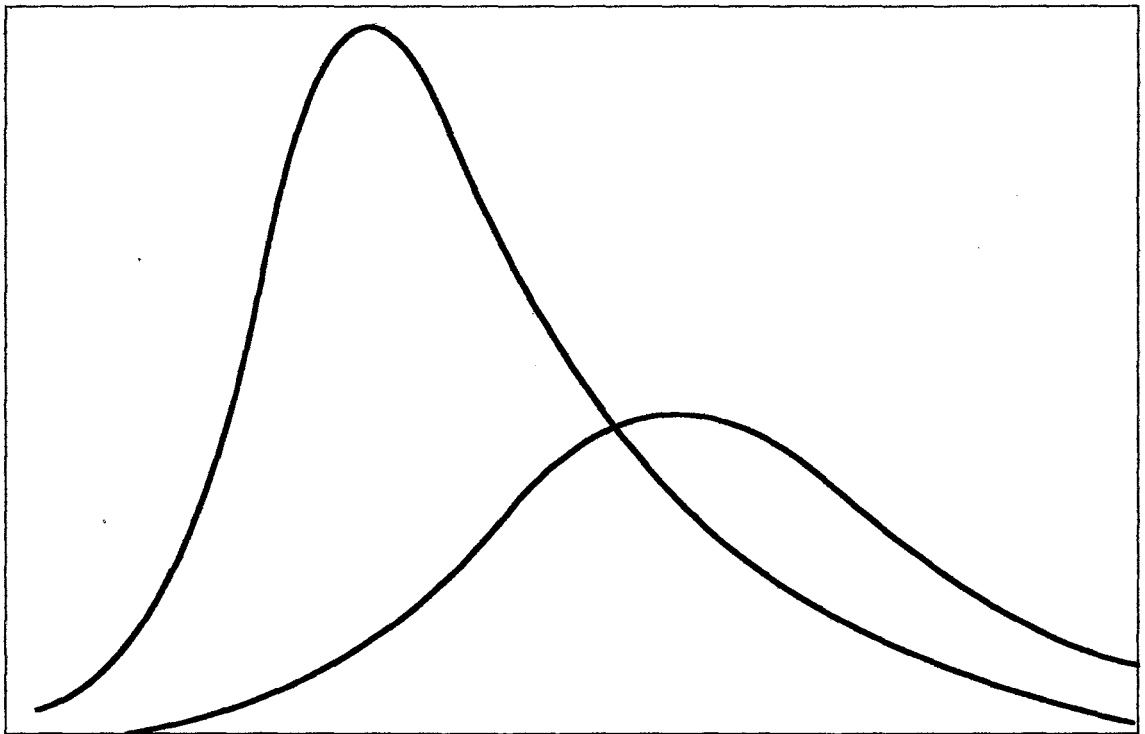


UNIVERSITAS GADJAH MADA  
PUSAT ANTAR UNIVERSITAS  
ILMU TEKNIK

Pemakaian Metode Sapuan–Ganda  
Untuk Penyelesaian  
Aliran Tak Permanen 1–D



oleh:  
Ir. Djoko Luknanto M.Sc., Ph.D.

Juni 1992

# PENGANTAR

Aliran tak permanen merupakan sifat aliran yang banyak dijumpai di sungai. Untuk keperluan perencanaan, pengendalian banjir dan pemeliharaan serta pekerjaan lainnya yang berhubungan dengan sungai, diperlukan hitungan aliran tak permanen. Buku ini membahas hitungan aliran tak permanen satu dimensi secara singkat dan jelas.

Metoda yang digunakan dalam hitungan ini adalah metoda numeris. Di dalam metoda numeris banyak teknik hitungan yang dapat digunakan dalam menghitung aliran tak permanen. Buku ini hanya membahas teknik numeris 'sapan-ganda' untuk menghitung aliran tak permanen. Teknik ini dipilih karena telah berhasil dipraktekkan di lapangan dan merupakan teknik numeris yang efisien dan hemat memori komputer, sehingga cocok untuk Indonesia.

Di dalam buku ini bahasan hanya mencakup dasar-dasar serta pemakaian teknik numeris 'sapan-ganda' untuk saluran atau sungai tunggal. Bahasanya disampaikan dengan sederhana dan jelas, sehingga pembaca hanya memerlukan pengetahuan matematika setingkat  $S_1$  untuk memahaminya. Buku ini tepat untuk praktisi yang bekerja di lapangan maupun bagi yang ingin berkenalan dengan teknik numeris hitungan aliran tak permanen.

Buku ini tidak dimaksudkan untuk memberi dasar yang kuat tentang teknik numeris ataupun hidrolika numerik, tetapi memberikan teknik numeris yang dapat diandalkan untuk menghitung aliran tak permanen di lapangan. Bagi peminat hidrolika numerik disarankan untuk membaca buku-buku pegangan yang ada. Perlu diketahui bahwa kuliah tentang hidrolika numerik di dalam dan luar negeri diberikan dalam waktu satu semester kuliah (lebih kurang 6 bulan lamanya), sehingga buku sesingkat ini tidak akan dapat menggantikannya. Bagi para pemodel pemakai teknik numeris yang dibahas dalam buku ini diharapkan menyadari bahwa untuk menjadi pemodel yang baik diperlukan pengetahuan dasar dan menyeluruh mengenai analisis numerik maupun hidrolika numerik.

Penyusun berharap buku yang singkat ini dapat memberikan sumbangan berharga untuk pembangunan Indonesia.

Yogyakarta, 1 Juni 1991

Djoko Luknanto

# DAFTAR ISI

halaman

PENDAHULUAN .....	1
PERSAMAAN KONTINYUITAS DAN MOMENTUM .....	1
DISKRITISASI PERSAMAAN DASAR .....	6
PERSAMAAN KERJA .....	8
METODE 'SAPUAN-GANDA' .....	10
KONDISI AWAL .....	12
KONDISI BATAS .....	12
RUANG $x-t$ .....	16
KONDISI BATAS INTERNAL .....	17
KESINGULERAN .....	23
CONTOH SOAL .....	24
CARA PENYELESAIAN .....	26
BEBERAPA HASIL SIMULASI .....	26
PERTEMUAN BEBERAPA SALURAN .....	32
CONTOH MASUKAN DAN KELUARAN .....	35
PROGRAM KOMPUTER ATP1DUST .....	41
STRUKTUR DATA MASUKAN .....	62
DAFTAR PUSTAKA .....	67

## PENDAHULUAN

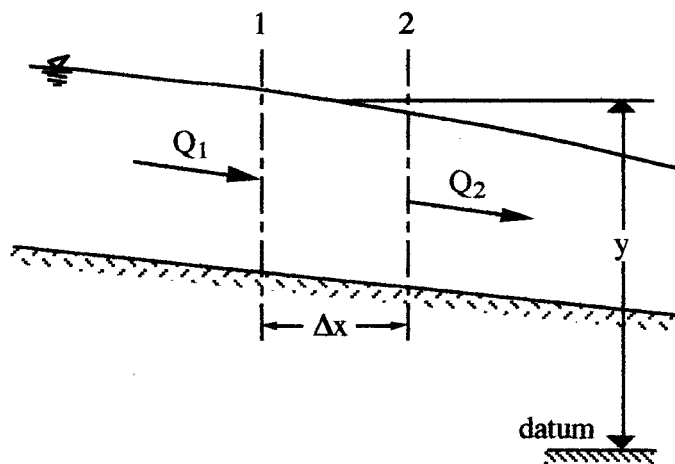
Pada umumnya hukum alam yang mengatur aliran tak permanen satu dimensi pada saluran/sungai alami ada dua macam yaitu hukum konservasi massa dan momentum. Kedua hukum tersebut biasa dinyatakan dalam persamaan matematis yang dikenal dengan nama persamaan kontinuitas dan dinamik. Kedua persamaan tersebut sulit dicari penyelesaian analitisnya, oleh karena itu untuk pemakaian praktis di lapangan kedua persamaan matematis tersebut diselesaikan secara numeris. Dalam buku ini dijelaskan secara rinci salah satu teknik numeris yang dapat digunakan untuk menyelesaikan persamaan tersebut, teknik numeris tersebut lazim disebut metoda 'double-sweep' atau '*sapuan-ganda*.'

Buku ini terutama dimaksudkan untuk memberikan cara hitungan aliran tak permanen pada saluran tunggal yang dapat dilengkapi dengan beberapa bangunan air antara lain kantong banjir, bendung, pintu sorong, bangunan pengambilan dlsb. Bangunan air yang diberikan di dalam buku ini hanya sebagai contoh. Diharapkan dari contoh-contoh ini pembaca dapat mengembangkan sendiri seandainya dibutuhkan suatu bangunan baru yang tidak dibahas di dalam buku ini.

Pada bagian akhir dari buku ini diberikan contoh-contoh hitungan dengan program komputer dalam bahasa FORTRAN. Program komputernya disertakan pula pada bagian akhir dari buku ini.

## PERSAMAAN KONTINYUITAS DAN MOMENTUM

Persamaan kontinuitas untuk aliran tak permanen satu dimensi dapat dijabarkan dengan pertolongan sebuah volume kontrol seperti yang tertera pada Gambar 1. Volume kontrol adalah pias air yang di-isolasi dari sekelilingnya sehingga dapat diamati secara rinci semua debit yang masuk dan keluar.



Gambar 1. Volume kontrol untuk penjabaran persamaan kontinuitas

Ditinjau pias air sepanjang  $\Delta x$  seperti tampak dalam Gambar 1. Pada pengaliran muka air bebas  $Q_2$  tidaklah perlu sama dengan  $Q_1$ , sehingga perbedaan tersebut dapat dinyatakan dalam persamaan sbb:

$$Q_2 - Q_1 = \frac{\partial Q}{\partial x} \Delta x \quad (1)$$

dimana  $Q_1$  adalah debit masuk volume kontrol dan  $Q_2$  adalah debit keluar volume kontrol,  $\Delta x$  adalah panjang pias volume kontrol, dan  $\frac{\partial Q}{\partial x}$  adalah ~~kecepatan~~ <sup>kecepatan</sup> perubahan nilai  $Q$  sepanjang  $\Delta x$ .

Dalam Pers. (1), jika debit lebih banyak yang masuk volume kontrol ( $Q_1 > Q_2$ ), maka nilai  $\frac{\partial Q}{\partial x}$  adalah negatif, sedangkan jika debit lebih banyak yang keluar volume kontrol ( $Q_2 > Q_1$ ), maka nilai  $\frac{\partial Q}{\partial x}$  adalah positif.

Karena sepanjang  $\Delta x$  mungkin terjadi penambahan atau pengurangan debit, jadi luas tampang basah pada pias tersebut dapat berubah pula untuk mengimbangi perubahan debit tersebut. Besarnya perubahan tersebut sepanjang  $\Delta x$  dapat dinyatakan dalam persamaan sbb:

$$\frac{\partial A}{\partial t} \Delta x \quad (2)$$

Dalam Pers. (2), jika debit lebih banyak yang masuk volume kontrol ( $Q_1 > Q_2$ ), maka nilai  $\frac{\partial A}{\partial t}$  adalah positif, sedangkan jika debit lebih banyak yang keluar volume kontrol ( $Q_2 > Q_1$ ), maka nilai  $\frac{\partial A}{\partial t}$  adalah negatif. Jadi kedua persamaan di atas, Pers. (1) dan (2), harus mempunyai nilai yang sama tetapi berlawanan tandanya, sehingga didapat persamaan kontinuitas untuk aliran tidak permanen sebagai berikut:

$$\frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} = 0 \quad (3)$$

Jika terdapat aliran dari samping sepanjang pias  $\Delta x$ , maka Pers. (3) menjadi:

$$\frac{\partial Q}{\partial x} + \frac{\partial A}{\partial t} = q_{\text{lateral}} \quad (4)$$

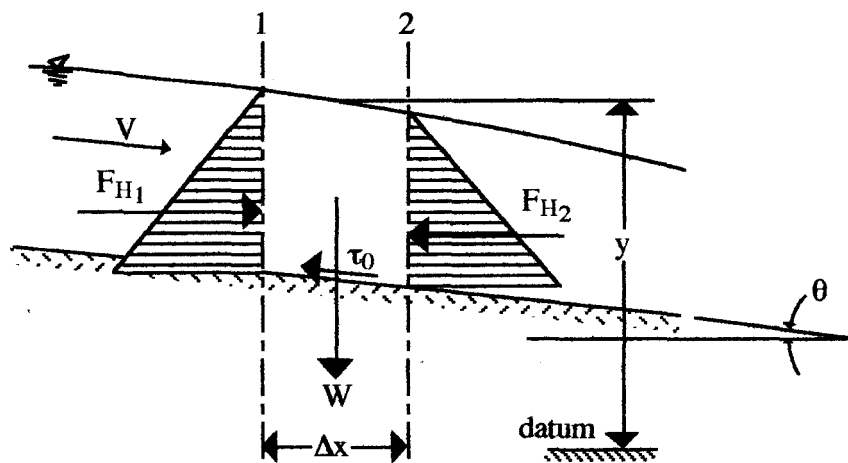
dimana  $Q$  adalah debit,  $A$  adalah luas tampang basah,  $q_{\text{lateral}}$  adalah debit samping per unit panjang saluran ( $q_{\text{lateral}}$  positif menandakan masuk ke volume kontrol),  $x$  dan  $t$  adalah menunjukkan panjang dan waktu.

Selain hukum kekekalan massa, suatu aliran air harus memenuhi hukum kekekalan momentum. Hukum kekekalan momentum yang dinyatakan dalam persamaan momentum sebenarnya adalah penjabaran dari gaya-gaya dan momentum yang bekerja pada air dalam

volume kontrol, sehingga menyebabkan air tersebut mengalir. Hukum kekekalan momentum mengatakan bahwa

“jumlah fluks momentum yang masuk dan keluar volume kontrol + jumlah gaya-gaya yang bekerja pada volume kontrol = perubahan momentum didalam volume kontrol.”

Untuk menerangkan dan menerapkan hukum kekekalan momentum di atas, maka digunakan lagi konsep volume kontrol seperti terlihat dalam Gambar 2.



Gambar 2. Gaya-gaya yang bekerja pada sebuah volume kontrol.

Momentum ( $M$ ) dalam suatu volume kontrol adalah perkalian antara massa ( $m$ ) dan kecepatan ( $V$ ), kalau dinyatakan dalam persamaan:

$$M = m \times V$$

sedangkan fluks momentum adalah perkalian antara fluks massa ( $\rho VA$ ) kali kecepatan, kalau dinyatakan dalam persamaan:

$$\frac{\Delta M}{\Delta t} = \rho VA V = \rho V^2 A$$

Jika digunakan anggapan: (1) aliran satu dimensi sehingga kecepatan aliran untuk setiap titik pada luas tampang basah sama nilainya dan (2) rapat massa  $\rho$  adalah konstan, maka perubahan momentum didalam volume kontrol:

$$\frac{\partial}{\partial t} (\rho A \Delta x V) = \rho \frac{\partial (AV)}{\partial t} \Delta x \quad (5)$$

sedangkan fluks momentum yang masuk dan keluar volume kontrol

$$\rho V^2 A - \left[ \rho V^2 A + \frac{\partial}{\partial x} (\rho V^2 A) \Delta x \right] = - \rho \frac{\partial}{\partial x} (V^2 A) \Delta x \quad (6)$$

Gaya-gaya yang bekerja pada volume kontrol adalah gaya berat, gaya gesekan dan gaya hidrostatika. Masing-masing gaya ini akan dibahas pada bab berikut.

Jika dipakai anggapan bahwa kemiringan dasar saluran adalah kecil atau dengan perkataan lain sudut  $\theta$  (lihat Gambar 2) adalah kecil, sehingga

$$\theta \approx 0 \Rightarrow \cos \theta \approx 1, \text{ maka } \sin \theta = \cos \theta \tan \theta \approx \tan \theta$$

Komponen gaya berat air yang mendorong air dapat dinyatakan sebagai:

$$W \sin \theta = \rho g A \Delta x \sin \theta = \rho g A \Delta x \tan \theta = \rho g A S_0 \Delta x \quad (7)$$

dimana  $S_0$  adalah sudut kemiringan dasar saluran.

Jika dipakai anggapan bahwa gaya gesekan pada aliran tak permanen masih mengikuti hukum-hukum untuk aliran permanen, maka gaya gesekan dapat dinyatakan sebagai

$$\begin{aligned} P \Delta x \tau_0 &= P \Delta x \rho g R S_f \\ &= \rho g A S_f \Delta x \end{aligned} \quad (8)$$

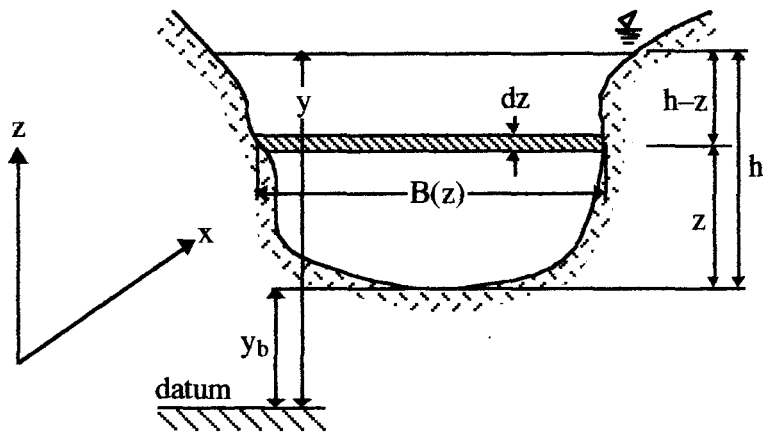
dimana  $P$  adalah keliling basah,  $\tau_0$  adalah tegangan gesek pada keliling basah,  $R$  adalah radius hidrolis tampang basah ( $= A/P$ ), dan  $S_f$  adalah garis kemiringan energi.

Jika dipakai anggapan bahwa percepatan vertikal aliran dapat diabaikan, maka tekanan dalam aliran adalah tekanan hidrostatika. Penjabaran gaya hidrostatika yang bekerja pada volume kontrol diperlihatkan pada Gambar 3. Gaya hidrostatika yang bekerja pada tampang lintang saluran dapat dinyatakan sebagai

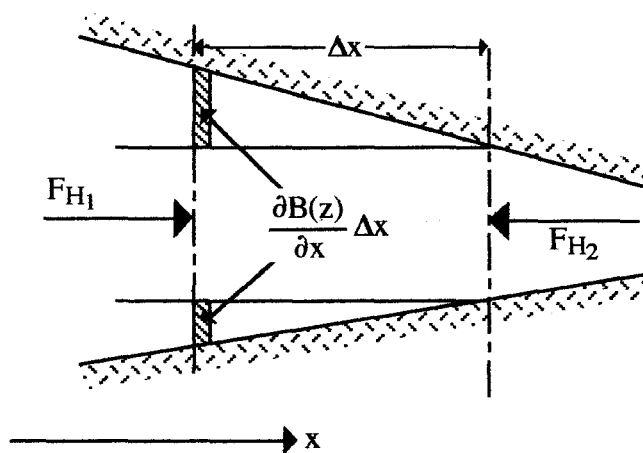
$$F_H = \int_{z=0}^{z=h} \rho g (h-z) B(z) dz \quad (9)$$

sehingga gaya hidrostatika total adalah

$$\begin{aligned} \Delta F_H &= F_{H1} - F_{H2} \\ &= F_H - \left( F_H + \frac{\partial F_H}{\partial x} \Delta x \right) \\ &= -\rho g \frac{\partial}{\partial x} \int_{z=0}^{z=h} (h-z) B(z) dz \Delta x \\ &= -\rho g \left[ \frac{\partial h}{\partial x} \int_{z=0}^{z=h} B(z) dz + \int_{z=0}^{z=h} (h-z) \frac{\partial B(z)}{\partial x} \Big|_{h=\text{konst}} dz \Delta x \right] \\ &\quad \underbrace{\hspace{10em}}_{\text{luas tampang basah}} \end{aligned}$$



(a) Tampang lintang volume kontrol



(b) Gaya-gaya hidrostatis dan gaya dinding saluran

Gambar 3. Tampang lintang, gaya hidrostatis dan gaya dinding

$$\Delta F_H = -\rho g \frac{\partial h}{\partial x} A \Delta x - \rho g \int_{z=0}^{z=h} (h-z) \frac{\partial B(z)}{\partial x} \Big|_{h=\text{konst}} dz \Delta x \quad (10)$$

suku terakhir

Untuk saluran prisma suk terakhir dari Pers. (10) nilainya mendekati nol sehingga diabaikan. Untuk saluran tak prisma yang perubahannya  $B(z)$  tidak mendadak, maka suku terakhir ini merupakan gaya yang menekan pada dinding saluran, sehingga dinding saluran memberi reaksi yang besarnya sama dengan arah yang berlawanan (lihat Gambar 3). Jadi baik untuk saluran prisma maupun tidak, gaya hidrostatis yang bekerja pada volume



kontrol dapat dinyatakan sebagai

$$\Delta F_H = -\rho g \frac{\partial h}{\partial x} A \Delta x \quad (11)$$

Jika Pers. (5) s/d (11) ditambahkan akan didapat persamaan momentum sebagai berikut

$$\begin{aligned} \frac{\partial(AV)}{\partial t} &= -\frac{\partial(AV^2)}{\partial x} + gAS_0 - gAS_f - gA \frac{\partial h}{\partial x} \\ \frac{\partial Q}{\partial t} + \frac{\partial(AV^2)}{\partial x} + gA \left[ \frac{\partial h}{\partial x} - S_0 + S_f \right] &= 0 \\ \frac{\partial Q}{\partial t} + \frac{\partial(AV^2)}{\partial x} + gA \left[ \frac{\partial y}{\partial x} + \frac{-\partial y_b}{\partial x} - S_0 + S_f \right] &= 0 \\ \frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left( \frac{Q^2}{A} \right) + gA \left[ \frac{\partial y}{\partial x} + \underbrace{\frac{-\partial y_b}{\partial x}}_{S_0} - S_0 + S_f \right] &= 0 \end{aligned} \quad (12)$$

Jika digunakan koefisien Coriolis,  $\alpha$ , untuk mengoreksi pemakaian rumus kecepatan aliran rerata ( $V$ ) sehingga mewakili distribusi kecepatan disetiap titik didalam tampang basah aliran, maka Pers. (12) dapat ditulis sebagai

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left( \alpha \frac{Q^2}{A} \right) + gA \left[ \frac{\partial y}{\partial x} + S_f \right] = 0$$

Sehingga persamaan momentum dalam bentuknya yang terakhir, yang akan dipakai pada perhitungan selanjutnya, dapat ditulis sebagai

$$\underbrace{\frac{\partial Q}{\partial t}}_{\text{suku 1}} + \underbrace{2\alpha \frac{Q}{A} \frac{\partial Q}{\partial x}}_{\text{suku 2}} - \underbrace{\alpha \left[ \frac{Q}{A} \right]^2 \frac{\partial A}{\partial x}}_{\text{suku 3}} + \underbrace{gA \frac{\partial y}{\partial x}}_{\text{suku 4}} + \underbrace{gA S_f}_{\text{suku 5}} = 0 \quad (13)$$

dimana  $Q$  adalah debit,  $x$  adalah panjang saluran,  $A$  adalah luas tampang basah,  $t$  adalah waktu,  $\alpha$  adalah koefisien Coriolis,  $y$  adalah elevasi muka air,  $g$  adalah percepatan gravitasi dan  $S_f$  adalah kemiringan garis energi.

## DISKRITISASI PERSAMAAN DASAR

Persamaan dasar (3) dan (13) akan didiskritisasi untuk mendapatkan persamaan kerja yang cocok untuk digunakan dalam hitungan numerik. Dalam buku ini akan digunakan metoda beda hingga sebagai metoda numerisnya. Dalam metoda numeris beda hingga deret Taylor merupakan deret yang dipakai sebagai dasar untuk menurunkan skema-skema numerik. Ekspansi deret Taylor sekitar titik  $(x_i, t_n)$  dimana  $\Delta x = x_{i+1} - x_i$  dan  $\Delta t = t_{n+1} - t_n$  adalah sbb:

$$\begin{aligned} f(x+\Delta x) &= f(x_{i+1}) \\ &= f(x_i) + \left. \frac{\partial f}{\partial x} \right|_{x_i} \frac{\Delta x}{1!} + \left. \frac{\partial^2 f}{\partial x^2} \right|_{x_i} \frac{\Delta x^2}{2!} + \dots \end{aligned} \quad (14)$$

Jadi,

$$\left. \frac{\partial f}{\partial x} \right|_{x_i} = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} + O(\Delta x) \quad (15)$$

sedangkan turunan terhadap waktu,  $t$ , dapat ditulis sebagai

$$\left. \frac{\partial f}{\partial t} \right|_{t_n} = \frac{f(t_{n+1}) - f(t_n)}{\Delta t} + O(\Delta t) \quad (16)$$

Dari Pers. (15) dan (16) dapat dikembangkan persamaan yang lebih kompleks. Dalam buku ini dipakai skema empat-titik Preissmann sebagai berikut:

$$f_i^{n+1} = f_i^n + \Delta f_i \quad \text{and} \quad f_{i+1}^{n+1} = f_{i+1}^n + \Delta f_{i+1} \quad (17)$$

$$\begin{aligned} \frac{\partial f}{\partial x} &= \theta \left. \frac{\partial f}{\partial x} \right|_{t_{n+1}} + (1-\theta) \left. \frac{\partial f}{\partial x} \right|_{t_n} \\ &= \theta \frac{f_{i+1}^{n+1} - f_i^{n+1}}{\Delta x} + (1-\theta) \frac{f_{i+1}^n - f_i^n}{\Delta x} \\ &= \frac{\theta}{\Delta x} (\Delta f_{i+1} - \Delta f_i) + \frac{1}{\Delta x} (f_{i+1}^n - f_i^n) \end{aligned} \quad (18)$$

$$\begin{aligned} \frac{\partial f}{\partial t} &= \frac{\left. \frac{\partial f}{\partial t} \right|_i + \left. \frac{\partial f}{\partial t} \right|_{i+1}}{2} \\ &= \frac{(f_i^{n+1} - f_i^n) + (f_{i+1}^{n+1} - f_{i+1}^n)}{2\Delta t} \\ &= \frac{1}{2\Delta t} (\Delta f_i + \Delta f_{i+1}) \end{aligned} \quad (19)$$

$$\begin{aligned} f(x,t) &= \frac{\theta}{2} (f_i^{n+1} + f_{i+1}^{n+1}) + \frac{1-\theta}{2} (f_i^n + f_{i+1}^n) \\ &= \frac{\theta}{2} (\Delta f_i + \Delta f_{i+1}) + \frac{1}{2} (f_i^n + f_{i+1}^n) \end{aligned} \quad (20)$$

dimana  $0 \leq \theta \leq 1$  disebut 'faktor pemberat waktu.'

$$\begin{aligned} S_f &= \theta \left[ \beta \frac{Q_i^{n+1} |Q_i^{n+1}|}{[K_i^{n+1}]^2} + (1-\beta) \frac{Q_{i+1}^{n+1} |Q_{i+1}^{n+1}|}{[K_{i+1}^{n+1}]^2} \right] + \\ &\quad (1-\theta) \left[ \beta \frac{Q_i^n |Q_i^n|}{[K_i^n]^2} + (1-\beta) \frac{Q_{i+1}^n |Q_{i+1}^n|}{[K_{i+1}^n]^2} \right] \end{aligned} \quad (21)$$

dimana  $0 \leq \beta \leq 1$  disebut 'faktor pemberat ruang,'  $\Delta A = b \Delta y$  dan  $K = \frac{1}{n} \frac{A^{5/3}}{p^{2/3}} = k_s \frac{A^{5/3}}{p^{2/3}}$ ,  $k_s$  = koefisien kekasaran Strickler,  $n$  = koefisien kekasaran Manning.

Dalam Pers. (17) s/d (20) parameter  $f$  dapat mewakili debit ( $Q$ ), elevasi muka air ( $y$ ), luas tampang basah ( $A$ ), faktor pengangkut ( $K$ ), dalam air ( $h$ ), dan lebar muka air ( $b$ ).

Ada satu teknik deret yang sering digunakan untuk melinierkan persamaan numerik, deret tersebut adalah:

$$\frac{1}{1+x} = 1 - x + \dots \quad \text{untuk } |x| \leq 1$$

sebagai contoh:

$$\begin{aligned} \frac{1}{A_i^n + \Delta A_i} &= \frac{1}{A_i^n \left[ 1 + \frac{\Delta A_i}{A_i^n} \right]} = \frac{1}{A_i^n} \left[ 1 - \frac{\Delta A_i}{A_i^n} \right] \\ &= \frac{A_i^n - \Delta A_i}{[A_i^n]^2} \end{aligned} \quad (22)$$

Diskritisasi persamaan kontinuitas, Pers. (3), dan persamaan momentum, Pers. (13), menggunakan skema empat titik Preissmann, Pers. (17) s/d (22) dijelaskan secara rinci pada bab berikut.

## PERSAMAAN KERJA

Dengan skema Preissmann seperti yang disajikan dalam Pers.(18) s/d (22) di atas, persamaan dasar Pers.(3) dan (13) dapat dirubah menjadi satu set persamaan linier yang mempunyai variabel yang tidak diketahui  $\Delta y_i$ ,  $\Delta Q_i$ ,  $\Delta y_{i+1}$ , dan  $\Delta Q_{i+1}$ . Sistem persamaan linier yang dihasilkan disajikan secara rinci sebagai berikut:

### Persamaan kontinuitas:

$$A \Delta y_{i+1} + B \Delta Q_{i+1} = C \Delta y_i + D \Delta Q_i + G \quad (23)$$

dimana:

$$A = \frac{b_{i+1}}{2\Delta t}, \quad B = D = \frac{\theta}{\Delta x}, \quad C = \frac{-b_i}{2\Delta t} \quad \text{dan} \quad G = \frac{Q_i - Q_{i+1}}{\Delta x} \quad (24)$$

### Persamaan dinamik:

$$AA \Delta y_{i+1} + BB \Delta Q_{i+1} = CC \Delta y_i + DD \Delta Q_i + GG \quad (25)$$

dimana koefisien AA, BB, CC, DD dan GG adalah sbb:

**AA merupakan jumlah dari:**

$$\text{suku 1: } 0 \quad (26.a)$$

$$\text{suku 2: } \frac{\alpha\theta b_{i+1}Q_{i+1}}{\Delta x A_{i+1}^2} [Q_i - Q_{i+1}] \quad (26.b)$$

$$\text{suku 3: } -\frac{\alpha\theta b_{i+1}}{4\Delta x} \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \left( \frac{2A_i}{A_{i+1}} - 1 \right) \right] \quad (26.c)$$

$$\text{suku 4: } \frac{\theta g}{2\Delta x} [A_i + A_{i+1} + b_{i+1} (y_{i+1} - y_i)] \quad (26.d)$$

$$\begin{aligned} \text{suku 5: } & \frac{\theta g b_{i+1}}{2} \left[ (1-\beta) \frac{Q_{i+1}|Q_{i+1}|}{K_{i+1}^2} + \beta \frac{Q_i|Q_i|}{K_i^2} \right] - \\ & (1-\beta) \theta g \frac{\partial K}{\partial y} \Big|_{i+1} (A_i + A_{i+1}) \frac{Q_{i+1}|Q_{i+1}|}{K_{i+1}^3} \end{aligned} \quad (26.e)$$

**BB merupakan jumlah dari:**

$$\text{suku 1: } \frac{0.5}{\Delta t} \quad (27.a)$$

$$\text{suku 2: } \frac{\alpha\theta}{\Delta x} \left[ \frac{Q_i}{A_i} - \frac{Q_i}{A_{i+1}} + \frac{2Q_{i+1}}{A_{i+1}} \right] \quad (27.b)$$

$$\text{suku 3: } \frac{\alpha\theta}{2\Delta x} \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \left[ \frac{A_i}{A_{i+1}} - 1 \right] \quad (27.c)$$

$$\text{suku 4: } 0 \quad (27.d)$$

$$\text{suku 5: } (1-\beta)\theta g (A_i + A_{i+1}) \frac{|Q_{i+1}|}{K_{i+1}^2} \quad (27.e)$$

**CC merupakan jumlah dari:**

$$\text{suku 1: } 0 \quad (28.a)$$

$$\text{suku 2: } \frac{\alpha\theta b_i Q_i}{\Delta x A_i^2} [Q_{i+1} - Q_i] \quad (28.b)$$

$$\text{suku 3: } \frac{\alpha\theta b_i}{4\Delta x} \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \left[ \frac{Q_i}{A_i} \left( 1 - \frac{2A_{i+1}}{A_i} \right) - \frac{Q_{i+1}}{A_{i+1}} \right] \quad (28.c)$$

$$\text{suku 4: } \frac{\theta g}{2\Delta x} [A_i + A_{i+1} + b_i (y_i - y_{i+1})] \quad (28.d)$$

$$\text{suku 5: } -\frac{\theta g b_i}{2} \left[ (1-\beta) \frac{Q_{i+1}|Q_{i+1}|}{K_{i+1}^2} + \beta \frac{Q_i|Q_i|}{K_i^2} \right] + \beta \theta g \frac{\partial K}{\partial y} \Big|_i [A_i + A_{i+1}] \frac{Q_i|Q_i|}{K_i^3} \quad (28.e)$$

DD merupakan jumlah dari:

$$\text{suku 1: } \frac{-0.5}{\Delta t} \quad (29.a)$$

$$\text{suku 2: } \frac{\alpha\theta}{\Delta x} \left[ \frac{2Q_i}{A_i} - \frac{Q_{i+1}}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \quad (29.b)$$

$$\text{suku 3: } \frac{\alpha\theta}{2\Delta x} \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \left[ \frac{A_{i+1}}{A_i} - 1 \right] \quad (29.c)$$

$$\text{suku 4: } 0 \quad (29.d)$$

$$\text{suku 5: } -\beta\theta g (A_i + A_{i+1}) \frac{|Q_i|}{K_i^2} \quad (29.e)$$

GG merupakan jumlah dari:

$$\text{suku 1: } 0 \quad (30.a)$$

$$\text{suku 2: } \frac{\alpha}{\Delta x} (Q_i - Q_{i+1}) \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right] \quad (30.b)$$

$$\text{suku 3: } \frac{\alpha}{4 \Delta x} (A_{i+1} - A_i) \left[ \frac{Q_i}{A_i} + \frac{Q_{i+1}}{A_{i+1}} \right]^2 \quad (30.c)$$

$$\text{suku 4: } \frac{g}{2\Delta x} [A_i + A_{i+1}] [y_i - y_{i+1}] \quad (30.d)$$

$$\text{suku 5: } -\frac{g}{2} (A_i + A_{i+1}) \left[ (1-\beta) \frac{Q_{i+1}|Q_{i+1}|}{K_{i+1}^2} + \beta \frac{Q_i|Q_i|}{K_i^2} \right] \quad (30.e)$$

## METODE 'SAPUAN-GANDA'

Persamaan kerja dari metoda 'sapan-ganda' adalah Pers. (23) dan (25) untuk  $i = 1, \dots, N-1$  dengan 'variabel tak diketahui' adalah  $\Delta y_i$  dan  $\Delta Q_i$  untuk  $i = 1, \dots, N$ . Dengan demikian terdapat  $2N$  variabel tak diketahui dengan  $2(N-1) = 2N-2$  persamaan, sehingga untuk menyelesaikan sistem persamaan linier, Pers. (23) dan (25) masih dibutuhkan tambahan 2 persamaan. Dua persamaan tambahan tersebut didapat dari dua kondisi batas hulu dan hilir.

Untuk memulai hitungan dibutuhkan pula kondisi awal berupa  $y_i$  dan  $Q_i$  untuk  $i = 1, \dots, N$ .

Sistem persamaan linier di atas dapat diselesaikan dengan sembarang 'linear solver' karena bentuknya secara umum dapat ditulis sebagai  $[A]\{\Delta\} = \{B\}$ . Tetapi penyelesaian general dengan 'linear solver package' biasanya membutuhkan memori yang besar dan waktu yang dibutuhkan untuk menyelesaikan persamaan di atas relatif lama. Oleh karena itu

di sini akan dibahas salah satu cara penyelesaian tanpa menggunakan matrik yaitu metoda 'sapan-ganda' yang akan dijelaskan di bawah ini.

1. Eliminasi  $\Delta Q_i$  dari Pers. (23) dan (25) menghasilkan:

$$\Delta y_i = L_i \Delta y_{i+1} + M_i \Delta Q_{i+1} + N_i \quad (31)$$

2. Diajukan suatu korelasi sbb:

$$\Delta Q_i = E_i \Delta y_i + F_i \quad (32)$$

3. Substitusi Pers. (31) dan (32) kedalam Pers. (23) akan menghasilkan persamaan berbentuk

$$\Delta Q_{i+1} = E_{i+1} \Delta y_{i+1} + F_{i+1}$$

dimana

$$E_{i+1} = \frac{L_i(C+DE_i)-A}{B-M_i(C+DE_i)} \quad (33.a)$$

$$F_{i+1} = \frac{N_i(C+DE_i)+DF_i+G}{B-M_i(C+DE_i)} \quad (33.b)$$

$$L_i = \frac{A(DD) - (AA)D}{C(DD) - (CC)D} \quad (33.c)$$

$$M_i = \frac{B(DD) - (BB)D}{C(DD) - (CC)D} \quad (33.d)$$

$$N_i = \frac{D(GG) - (DD)G}{C(DD) - (CC)D} \quad (33.e)$$

Tampak di atas bahwa Pers. (32) s/d (33) mempunyai hubungan 'recursive' dimana koefisien pengaruh,  $E_{i+1}$  dan  $F_{i+1}$ , nilainya tergantung dari nilai  $E_i$  dan  $F_i$ , sehingga koefisien pengaruh dapat dihitung untuk masing-masing titik-titik hitungan,  $i$ , asalkan koefisien pengaruh untuk  $i = 1$  telah dihitung terlebih dahulu. Inilah yang disebut dengan 'sapan ke hilir' dimana  $E_1$  dan  $F_1$  harganya dihitung dari kondisi batas hulu, kemudian semua koefisien pengaruh yang lainnya dapat dihitung dengan Pers. (33). Disamping itu koefisien pengaruh yang lain yaitu  $L_i$ ,  $M_i$ ,  $N_i$  dihitung untuk setiap titik-titik hitungan. Koefisien ini akan digunakan pada 'sapan ke hulu' yang akan dijelaskan di bawah ini.

Setelah semua koefisien pengaruh terhitung, maka akan dilakukan ‘sapuan ke hulu’ dimana  $\Delta y_N$  dan  $\Delta Q_N$  dihitung dari kondisi batas hilir. Setelah itu  $\Delta y_i$  dan  $\Delta Q_i$  untuk setiap titik-titik hitungan dapat dihitung mundur ke arah hilir dengan Pers. (31) dan (32).

Untuk memperjelas konsep dari metoda ‘sapuan-ganda,’ maka bagan alirnya diperlihatkan pada Gambar 4.

## KONDISI AWAL

Seperti telah dijelaskan di atas, untuk memulai hitungan ‘sapuan-ganda,’ diperlukan kondisi awal yang berupa nilai  $y_i$  dan  $Q_i$  untuk seluruh panjang sungai atau untuk  $i = 1$  s/d  $N$ .

## KONDISI BATAS

Dua kondisi batas masing-masing di hulu dan hilir saluran dibutuhkan untuk melengkapi persamaan dinamik dan kontinuitas, sehingga  $Q_i$  dan  $y_i$  untuk  $i = 1, \dots, N$  dapat dihitung untuk setiap ‘time step.’ Kondisi batas ini harus disesuaikan bentuknya sehingga sesuai dengan Pers. (32). Bentuk umum persamaan kondisi batas adalah sebagai berikut:

$$\alpha \Delta y_i + \beta \Delta Q_i = \gamma_i \text{ untuk } i=1 \text{ dan } N \quad (34)$$

Untuk memulai ‘sapuan ke hilir’ dibutuhkan nilai  $E_1$  dan  $F_1$  yang diperoleh dengan membandingkan Pers. (34) dengan Pers. (32) sehingga didapat hubungan

$$E_1 = -\frac{\alpha_1}{\beta_1} \text{ dan } F_1 = \frac{\gamma_1}{\beta_1} \quad (35)$$

dimana  $\alpha_1$ ,  $\beta_1$ , dan  $\gamma_1$  nilainya didapat dari kondisi batas hulu.

Untuk memulai ‘sapuan ke hulu,’ dipakai Pers. (34) dan (32) untuk nilai  $i = N$  yang dapat ditulis sebagai berikut:

$$\alpha_N \Delta y_N + \beta_N \Delta Q_N = \gamma_N \quad (36)$$

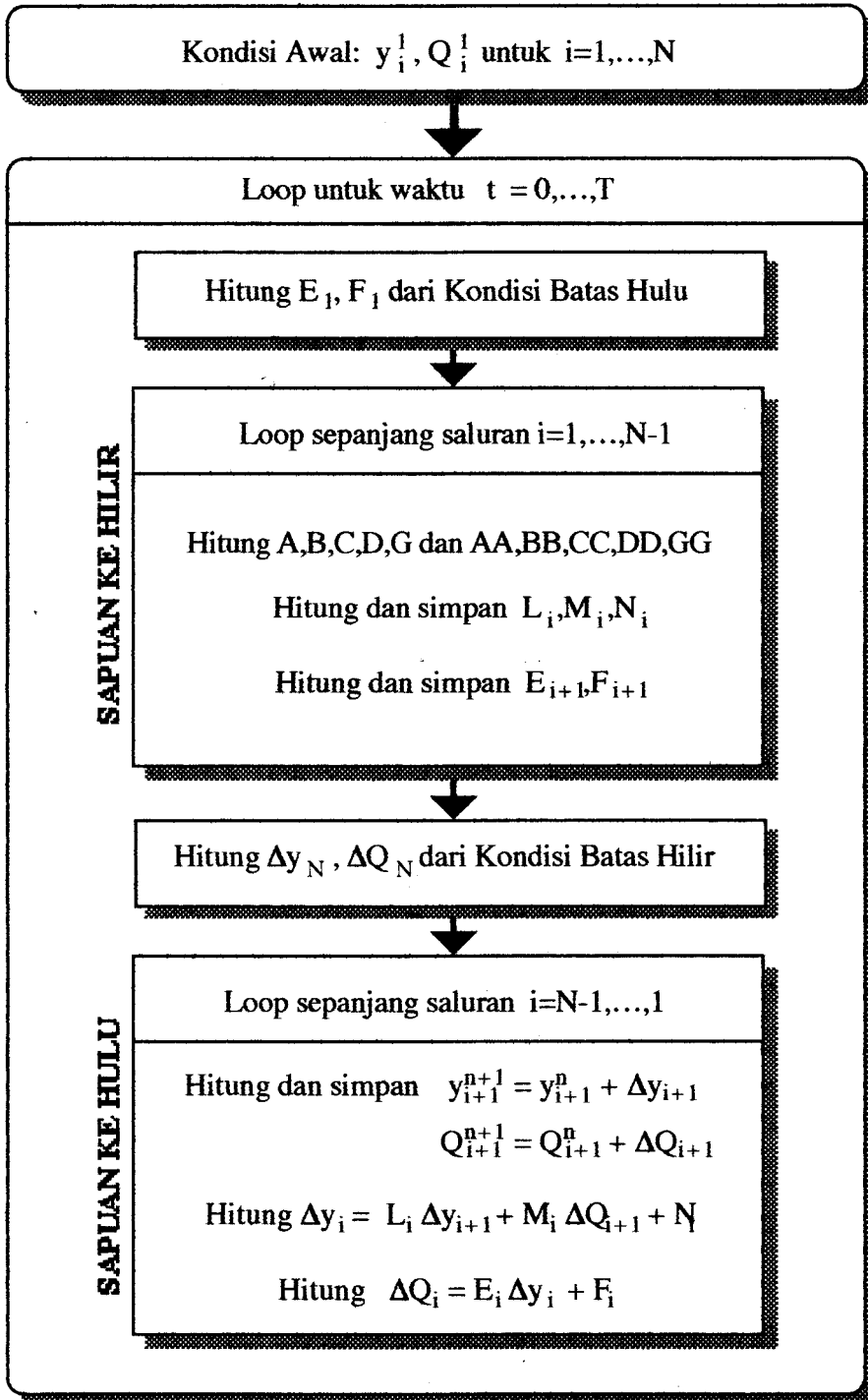
$$\Delta Q_N = E_N \Delta y_N + F_N \quad (37)$$

dengan menggunakan Pers. (36) dan (37) dapat dihitung nilai  $\Delta y_N$  dan  $\Delta Q_N$  sebagai berikut:

$$\Delta y_N = \frac{\gamma - \beta F_N}{\alpha + \beta E_N} \quad (38)$$

dan  $\Delta Q_N$  dapat dihitung dari Pers. (37) setelah  $\Delta y_N$  terhitung dari Pers. (38). Nilai  $\alpha_N$ ,  $\beta_N$ , dan  $\gamma_N$  didapat dari kondisi batas hilir, sedangkan  $E_N$  dan  $F_N$  didapat dari ‘sapuan ke hilir.’

**ALGORITMA 'SAPUAN-GANDA'  
UNTUK SALURAN TUNGGAL**



Gambar 4. Bagan Alir Metoda 'Sapuan-ganda'



Cara mendapatkan nilai  $\alpha$ ,  $\beta$ , dan  $\gamma$  untuk beberapa macam kondisi batas akan dijelaskan pada paragraf berikut.

Beberapa kondisi batas yang sering dijumpai dilapangan

•» **Q(t):** diketahui hidrograp.

Diinginkan pada setiap akhir 'time step' korelasi dalam hidrograp harus selalu dipenuhi.

$$Q^{n+1} = Q^n + \Delta Q \text{ atau}$$

$$Q^n + \Delta Q = Q(t_{n+1}) \text{ atau}$$

$$\Delta Q = Q(t_{n+1}) - Q^n$$

Jika dibandingkan dengan Pers. (34), maka didapat

$$\alpha = 0, \beta = 1, \gamma = Q(t_{n+1}) - Q^n \quad (39)$$

•» **Q = f(y):** diketahui debit sebagai fungsi elevasi atau 'rating curve.'

$$Q^{n+1} = Q^n + \Delta Q = f(y^{n+1})$$

**f(y)** dilinierkan dengan menggunakan deret Taylor, dengan mengabaikan 'higher order term' sbb:

$$f(y^{n+1}) = f(y^n) + \left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y$$

jadi

$$Q^n + \Delta Q = f(y^n) + \left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y$$

sehingga di dapat bentuk terakhir sbb:

$$\left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y - \Delta Q = Q^n - f(y^n)$$

Jika dibandingkan dengan Pers. (34), maka didapat

$$\alpha = \left. \frac{\partial f}{\partial y} \right|_{y^n}, \beta = -1, \gamma = Q^n - f(y^n) \quad (40)$$

•» **y(t):** diketahui elevasi muka air.

Sejalan dengan contoh pertama didapat:

$$y^{n+1} = y^n + \Delta y \text{ atau}$$

$$y^n + \Delta y = y(t_{n+1}) \text{ atau}$$

$$\Delta y = y(t_{n+1}) - y^n$$

Cara mendapatkan nilai  $\alpha$ ,  $\beta$ , dan  $\gamma$  untuk beberapa macam kondisi batas akan dijelaskan pada paragraf berikut.

Beberapa kondisi batas yang sering dijumpai dilapangan

•»  $Q(t)$ : diketahui hidrograp.

Diinginkan pada setiap akhir 'time step' korelasi dalam hidrograp harus selalu dipenuhi.

$$Q^{n+1} = Q^n + \Delta Q \text{ atau}$$

$$Q^n + \Delta Q = Q(t_{n+1}) \text{ atau}$$

$$\Delta Q = Q(t_{n+1}) - Q^n$$

Jika dibandingkan dengan Pers. (34), maka didapat

$$\alpha = 0, \beta = 1, \gamma = Q(t_{n+1}) - Q^n \quad (39)$$

•»  $Q = f(y)$ : diketahui debit sebagai fungsi elevasi atau 'rating curve.'

$$Q^{n+1} = Q^n + \Delta Q = f(y^{n+1})$$

$f(y)$  dilinierkan dengan menggunakan deret Taylor, dengan mengabaikan 'higher order term' sbb:

$$f(y^{n+1}) = f(y^n) + \left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y$$

jadi

$$Q^n + \Delta Q = f(y^n) + \left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y$$

sehingga di dapat bentuk terakhir sbb:

$$\left. \frac{\partial f}{\partial y} \right|_{y^n} \Delta y - \Delta Q = Q^n - f(y^n)$$

Jika dibandingkan dengan Pers. (34), maka didapat

$$\alpha = \left. \frac{\partial f}{\partial y} \right|_{y^n}, \beta = -1, \gamma = Q^n - f(y^n) \quad (40)$$

•»  $y(t)$ : diketahui elevasi muka air.

Sejalan dengan contoh pertama didapat:

$$y^{n+1} = y^n + \Delta y \text{ atau}$$

$$y^n + \Delta y = y(t_{n+1}) \text{ atau}$$

$$\Delta y = y(t_{n+1}) - y^n$$

Jika dibandingkan dengan Pers. (34), maka didapat

$$\alpha = 1, \beta = 0, \gamma = y(t_{n+1}) - y^n \quad (41)$$

Bahasan di bawah adalah mengenai cara penanganan secara rinci bagaimana 'sapuan-ganda' masih dapat digunakan walaupun  $\beta = 0$ . Bagi yang tidak tertarik dapat melewatinya dan langsung membaca bab berikutnya tanpa kehilangan bahasan pokok tentang aliran tak permanen.

Perlu mendapat perhatian karena nilai  $\beta = 0$ , maka  $E_1$  dan  $F_1$  tidak dapat dihitung, sehingga perlu ditangani secara khusus. Penanganannya adalah sebagai berikut :

1. 'sapuan ke hilir' tidak dimulai dari titik  $i=1$  tetapi dari titik  $i=2$ , sehingga  $E_1$  dan  $F_1$  tidak dibutuhkan karena 'sapuan ke hilir' dimulai dengan menghitung  $E_2$  dan  $F_2$  menggunakan Pers. (45).
2. 'sapuan ke hulu' dimulai seperti biasa, hanya pada titik  $i=1$  diperlukan hubungan baru yang mempunyai bentuk  $\Delta Q_1 = l_1 \Delta y_2 + m_1 \Delta Q_2 + n_1$ , sehingga  $\Delta Q_1$  dapat dihitung dari informasi di sebelah hilirnya. Koefisien pengaruh  $l_1$ ,  $m_1$ , dan  $n_1$  dihitung dengan Pers. (47).

Untuk keperluan tersebut di atas, maka persamaan kontinuitas dan dinamik ditulis lagi untuk titik 1 dan 2 sbb:

$$A \Delta y_2 + B \Delta Q_2 = C \Delta y_1 + D \Delta Q_1 + G \quad (42)$$

$$AA \Delta y_2 + BB \Delta Q_2 = CC \Delta y_1 + DD \Delta Q_1 + GG \quad (43)$$

dimana  $\Delta y_1$  diketahui nilainya sama dengan  $\gamma$ . Eliminasi  $\Delta Q_1$  dari persamaan di atas akan menghasilkan persamaan dalam bentuk

$$\Delta Q_2 = E_2 \Delta y_2 + F_2 \quad (44)$$

dimana

$$E_2 = \frac{A(DD) - D(AA)}{D(BB) - B(DD)} \quad \text{dan} \quad F_2 = \frac{D(CC \gamma + GG) - (DD)(C \gamma + G)}{D(BB) - B(DD)} \quad (45)$$

Eliminasi  $\Delta y_1$  dari Pers. (42) dan (43) di atas akan menghasilkan persamaan dalam bentuk

$$\Delta Q_1 = l_1 \Delta y_2 + m_1 \Delta Q_2 + n_1 \quad (46)$$

dimana

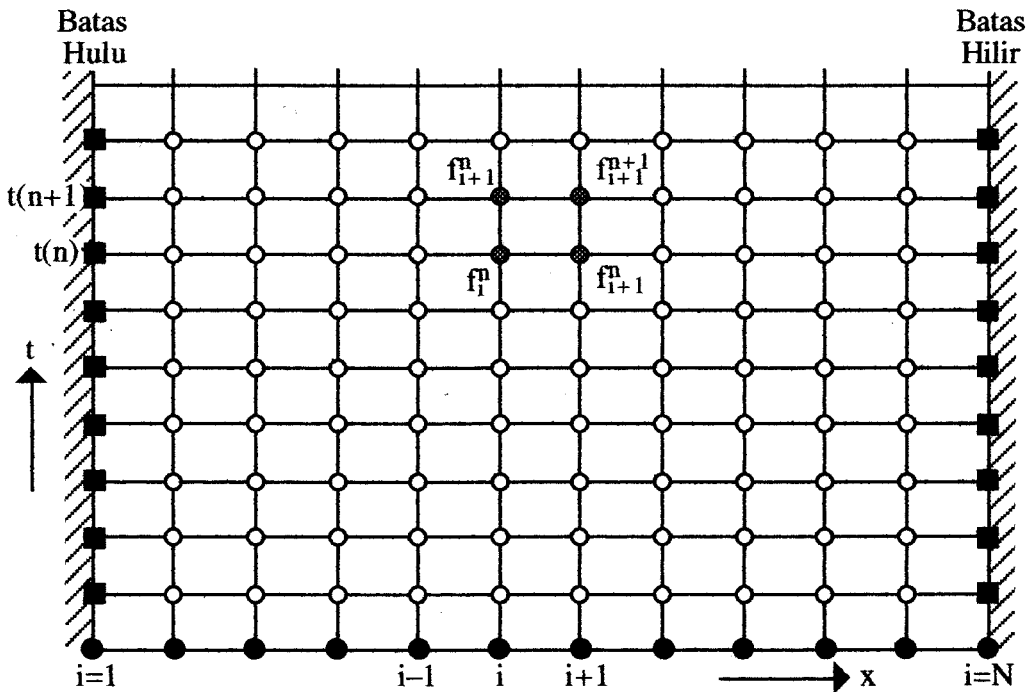
$$l_1 = \frac{A(CC) - C(AA)}{D(CC) - C(DD)} \quad \text{dan} \quad m_1 = \frac{B(CC) - C(BB)}{D(CC) - C(DD)} \quad \text{dan} \quad n_1 = \frac{C(GG) - G(CC)}{D(CC) - C(DD)} \quad (47)$$

Perlu mendapat perhatian bahwa semua koefisien ( $A, B, C, D, G, AA, BB, CC, DD, GG$  dll)

di atas dihitung untuk titik-titik hitungan  $i=1$  dan  $i=2$ .

## RUANG $x-t$

Didalam metoda beda hingga dalam menjelaskan skema yang dipakai biasanya digunakan ruang  $x-t$  untuk menggambarkan skema ybs (lihat Gambar 5). Skema diskritisasi Preissmann, yang disajikan dalam Pers. (17) s/d (21), biasa digolongkan kedalam skema empat titik, karena skema ini menggunakan informasi dari empat titik untuk mendiskritisasi persamaan dasar, Pers. (3) dan (13).



### KETERANGAN GAMBAR

Titik-titik yang harus diketahui datanya:

- Kondisi Batas
- Kondisi Awal

Titik-titik yang dihitung  $y$  dan  $Q$ -nya :

- Titik-titik hitungan
- Titik-titik hitungan yang dipakai dalam skema Preissmann

Gambar 5. Sket Ruang  $x-t$

Dengan menggunakan Gambar 5, mudah dijelaskan apa yang dimaksud dengan skema empat titik. Dalam mendiskritisasi persamaan dasar, maka pada titik hitungan  $i$  dipakai 4 titik yaitu  $(i,n)$ ,  $(i+1,n)$ ,  $(i,n+1)$  dan  $(i+1,n+1)$  (dalam Gambar 5 diberi simbol ●) dengan masing-masing titik hitungan mempunyai 2 variable tak diketahui yaitu  $(y,Q)$ , dimana dalam

Gambar 5 hanya dicantumkan sebagai  $f(x,t)$ .

Dalam Gambar 5 tampak bahwa titik-titik yang akan dihitung  $(y,Q)$ -nya adalah titik-titik 'interior' dalam ruang  $x-t$  (dalam Gambar 5 diberi simbol  $\circ$  ). Sedangkan di titik-titik hitungan pada batas hulu dan hilir (dalam Gambar 3 diberi simbol  $\blacksquare$  ) serta pada kondisi awal (dalam Gambar 5 diberi simbol  $\bullet$  )  $(y,Q)$ -nya harus telah diketahui nilainya.

## KONDISI BATAS INTERNAL

Didalam teknik numeris segala bangunan yang ada di sepanjang saluran disebut sebagai kondisi batas 'internal.' Pada prinsipnya semua kondisi batas internal harus dinyatakan dalam persamaan linier dengan bentuk seperti Pers. (23) dan (24) sbb:

$$A \Delta y_{i+1} + B \Delta Q_{i+1} = C \Delta y_i + D \Delta Q_i + G \quad (48)$$

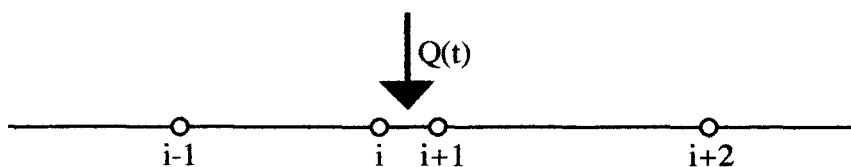
$$AA \Delta y_{i+1} + BB \Delta Q_{i+1} = CC \Delta y_i + DD \Delta Q_i + GG \quad (49)$$

Untuk menghasilkan dua persamaan linier seperti Pers. (48) dan (49) di atas dibutuhkan dua persamaan dasar yang diturunkan dari keadaan fisik kondisi batas 'internal' (bangunan) ybs. Kedua persamaan dasar tersebut adalah:

1. Hukum kontinuitas dan
2. Hukum-hukum fisika yang berlaku pada bangunan ybs.

### Contoh 1: 'Inflow' dari luar saluran

Dalam Gambar 6, perjanjian yang dipakai adalah  $Q(t)$  tandanya positif jika merupakan 'inflow' dan negatif jika 'outflow.' Jarak antara titik  $i$  dan  $i+1$  tidak mempunyai arti secara fisik, hanya sebagai alat bantu untuk melaksanakan teknik numeris.



Gambar 6. Saluran tunggal dengan inflow/outflow

Persamaan 1: kontinuitas

$$Q_i^{n+1} + Q(t_{n+1}) = Q_{i+1}^{n+1} \quad (50)$$

jika dibandingkan dengan Pers. (48), maka didapat:

$$A=C=0, B=D=1, G = Q_i^n + Q(t_{n+1}) - Q_{i+1}^n \quad (51)$$

Persamaan 2: Ada beberapa kemungkinan hukum fisika yang dapat diberlakukan disini.

- momentum akan bertambah jika  $Q(t)$  arahnya tidak tegak lurus saluran
- garis enersinya mempunyai elevasi yang sama pada titik  $i$  dan  $i+1$
- elevasi muka airnya sama pada titik  $i$  dan  $i+1$

Ternyata berdasarkan pengalaman, elevasi muka airnya dianggap sama sudah mencukupi untuk kepentingan praktis di lapangan. Jadi persamaan kedua menjadi:

$$\begin{aligned} y_i^{n+1} &= y_{i+1}^{n+1} \\ y_i^n + \Delta y_i &= y_{i+1}^n + \Delta y_{i+1} \end{aligned} \quad (52)$$

jika dibandingkan dengan Pers. (49), maka didapat:

$$AA=CC=1, BB=DD=0, GG = y_i^n - y_{i+1}^n \approx 0 \neq 0 \quad (53)$$

### Contoh 2: Kantong Banjir

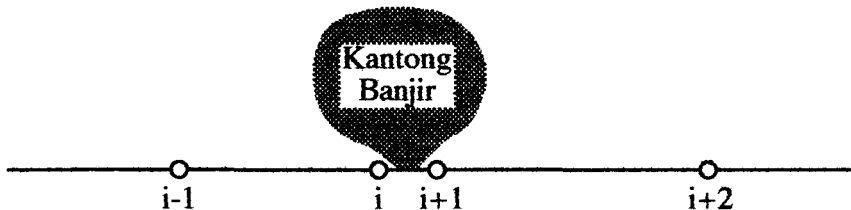
Dalam Gambar 7, perjanjian yang dipakai adalah  $Q_p$  tandanya positif jika masuk kekantong banjir. Asumsi: hubungan antara luas permukaan kantong banjir dengan elevasi muka air sudah diketahui, jadi  $S(y)$  telah tersedia.

Persamaan 1: Elevasi muka air dianggap sama antara titik  $i$  dan  $i+1$ , sehingga hasilnya seperti contoh sebelumnya:

$$\begin{aligned} y_i^{n+1} &= y_{i+1}^{n+1} \\ y_i^n + \Delta y_i &= y_{i+1}^n + \Delta y_{i+1} \end{aligned} \quad (54)$$

jika dibandingkan dengan Pers. (49), maka didapat:

$$AA=CC=1, BB=DD=0, GG = y_i^n - y_{i+1}^n \approx 0 \neq 0 \quad (55)$$



Gambar 7. Saluran dengan kantong banjir

Persamaan 2: kontinyuitas

$$Q_{i+1}^{n+1} = Q_i^{n+1} - Q_p^{n+1} \quad (56)$$

Sekarang dijabarkan hukum kontinuitas di dalam kantong banjir.

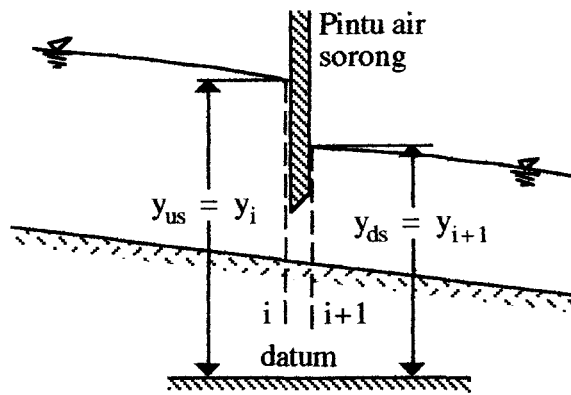
$$\begin{aligned} \overline{Q}_p \Delta t &= V(y_i^{n+1}) - V(y_i^n) \\ [\theta Q_p^{n+1} + (1-\theta)Q_p^n] \Delta t &= V(y_i^n) + \left. \frac{\partial V}{\partial y} \right|_{y_i^n} \Delta y_i + \dots - V(y_i^n) \\ \theta(Q_i^{n+1} - Q_{i+1}^{n+1}) + (1-\theta)(Q_i^n - Q_{i+1}^n) &= \frac{S(y_i)}{\Delta t} \Delta y_i \\ \theta(Q_i^n + \Delta Q_i - Q_{i+1}^n - \Delta Q_{i+1}) + (1-\theta)(Q_i^n - Q_{i+1}^n) &= \frac{S(y_i)}{\Delta t} \Delta y_i \\ \theta \Delta Q_i - \theta \Delta Q_{i+1} + (Q_i^n - Q_{i+1}^n) &= \frac{S(y_i)}{\Delta t} \Delta y_i \end{aligned} \quad (57)$$

jika dibandingkan dengan Pers. (48), maka didapat:

$$A = 0, C = \frac{S(y_i^n)}{\Delta t}, B = D = -\theta \text{ dan } G = Q_{i+1}^n - Q_i^n \quad (58)$$

### Contoh 3: Pintu Air

Pada pintu sorong seperti terlihat pada Gambar 8 persamaan kontinuitas dan debit melalui pintu sorong tersebut dapat digunakan untuk menghitung koefisien pengaruh 'sapan-ganda.' Persamaan tersebut diterangkan secara rinci di bawah ini.



Gambar 8. Saluran dengan pintu sorong

Persamaan 1: kontinuitas

$$\begin{aligned} Q_{i+1}^{n+1} &= Q_i^{n+1} \\ Q_{i+1}^n + \Delta Q_{i+1} &= Q_i^n + \Delta Q_i \end{aligned} \quad (59)$$

jika dibandingkan dengan Pers. (48), maka didapat:

$$A=C=0, B=D=1, G = Q_i^n - Q_{i+1}^n \quad (60)$$

Penjabaran hukum kontinuitas dilakukan seperti contoh-contoh sebelumnya. Untuk pintu sorong penjabarannya sebagai berikut:

Persamaan 2: Debit melalui pintu sorong

$$Q = C_D A \sqrt{2g(y_{us}-y_{ds})}$$

$$Q = C_D A \sqrt{2g} f(y_{us}, y_{ds})$$

$$Q_i^{n+1} = C_g f(y_{us}^{n+1}, y_{ds}^{n+1})$$

$$Q_i^n + \Delta Q_i = C_g \left[ f(y_{us}^n, y_{ds}^n) + \frac{\partial f}{\partial y_{us}} \Big|_{(y_{us}^n, y_{ds}^n)} \Delta y_{us} + \frac{\partial f}{\partial y_{ds}} \Big|_{(y_{us}^n, y_{ds}^n)} \Delta y_{ds} \right]$$

$$\text{Jadi AA} = C_g \frac{\partial f}{\partial y_{ds}} \Big|_{y_{ds}^n}, \text{ BB} = 0, \text{ CC} = -C_g \frac{\partial f}{\partial y_{us}} \Big|_{y_{us}^n}, \text{ DD} = 1, \text{ GG} = Q_i^n - C_g f(y_{us}^n, y_{ds}^n) \quad (62)$$

dimana

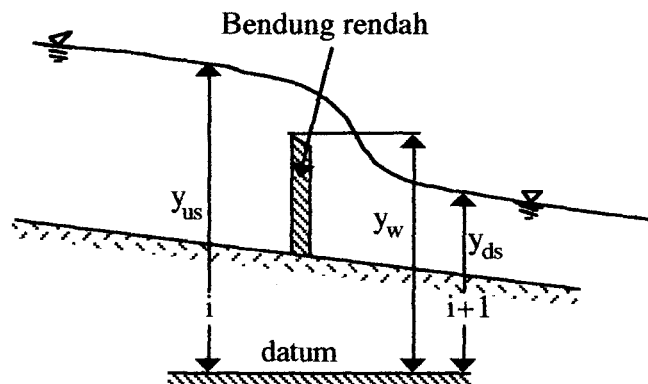
$$C_g = C_D A \sqrt{2g} \quad (63)$$

$$f(y_{us}, y_{ds}) = \sqrt{(y_{us} - y_{ds})} \quad (64)$$

$$\frac{\partial f}{\partial y_{us}} = \frac{1}{2\sqrt{(y_{us} - y_{ds})}} \quad (65)$$

$$\frac{\partial f}{\partial y_{ds}} = \frac{-1}{2\sqrt{(y_{us} - y_{ds})}} \quad (66)$$

#### Contoh 4: Bendung rendah segi empat



Gambar 9. Bendung rendah persegi empat

Persamaan 1: kontinuitas



$$\begin{aligned} Q_{i+1}^{n+1} &= Q_i^{n+1} \\ Q_{i+1}^n + \Delta Q_{i+1} &= Q_i^n + \Delta Q_i \end{aligned} \quad (67)$$

jika dibandingkan dengan Pers. (48), maka didapat:

$$A=C=0, B=D=1, G = Q_i^n - Q_{i+1}^n \quad (68)$$

Persamaan 2: Debit yang melalui bendung rendah tergantung dari dua macam keadaan aliran air di hulu bendung yaitu aliran dengan arus bebas dan arus menyelim.

Bendung dengan arus bebas terjadi jika:  $y_{ds} - y_w \leq 2/3 (y_{us} - y_w)$

$$Q = C_D B \frac{2}{3} \sqrt{\frac{2}{3} g} (y_{us} - y_w)^{1.5} \quad (69)$$

Bendung dengan arus menyelim terjadi jika:  $y_{ds} - y_w > 2/3 (y_{us} - y_w)$

$$Q = C_D B \sqrt{2 g} (y_{us} - y_{ds})^{0.5} (y_{ds} - y_w) \quad (70)$$

dimana  $Q$  adalah debit bendung,  $C_D$  adalah koefisien debit,  $B$  adalah lebar mercu bendung,  $y_{us}$  adalah elevasi muka air hulu,  $y_{ds}$  adalah elevasi muka air hilir, dan  $y_w$  adalah elevasi mercu bendung.

Secara umum persamaan debit bendung (69) dan (70) dapat ditulis sebagai

$$Q^{n+1} = f(y_{us}^{n+1}, y_{ds}^{n+1}) \quad (71)$$

sehingga didapat hubungan sbb:

$$Q^n + \Delta Q = f(y_{us}^n, y_{ds}^n) + \left. \frac{\partial f}{\partial y_{us}} \right]^n \Delta y_{us} + \left. \frac{\partial f}{\partial y_{ds}} \right]^n \Delta y_{ds} \quad (72)$$

Untuk  $Q_i$  positif artinya arah aliran adalah dari titik  $i$  ke titik  $i+1$ , maka Pers. (72) dapat ditulis sebagai

$$Q^n + \Delta Q = f(y_i^n, y_{i+1}^n) + \left. \frac{\partial f}{\partial y_{us}} \right]^n \Delta y_i + \left. \frac{\partial f}{\partial y_{ds}} \right]^n \Delta y_{i+1} \quad (73)$$

sehingga jika dibandingkan dengan Pers. (49) didapat

$$AA = \left. \frac{\partial f}{\partial y_{ds}} \right]_{y_{i+1}^n}, \quad BB = 0, \quad CC = - \left. \frac{\partial f}{\partial y_{us}} \right]_{y_i^n}, \quad DD = 1, \quad GG = Q_i^n - f(y_i^n, y_{i+1}^n) \quad (74)$$

Untuk  $Q_i$  negatif artinya arah aliran adalah dari titik  $i+1$  ke titik  $i$ , maka Pers. (72) dapat ditulis sebagai

$$Q^n + \Delta Q = f(y_i^n, y_{i+1}^n) + \left. \frac{\partial f}{\partial y_{us}} \right]^n \Delta y_{i+1} + \left. \frac{\partial f}{\partial y_{ds}} \right]^n \Delta y_i \quad (75)$$

sehingga jika dibandingkan dengan Pers. (49) didapat

$$AA = -\left. \frac{\partial f}{\partial y_{us}} \right]^n_{y_{i+1}^n}, \quad BB = 0, \quad CC = \left. \frac{\partial f}{\partial y_{ds}} \right]^n_{y_i^n}, \quad DD = 1, \quad GG = Q_i^n + f(y_i^n, y_{i+1}^n) \quad (76)$$

• Untuk bendung arus bebas:

$$\begin{aligned} f(y_{us}, y_{ds}) &= C_D B \frac{2}{3} \sqrt{\frac{2}{3} g} (y_{us} - y_w)^{1.5} \\ \frac{\partial f}{\partial y_{us}} &= C_D B \sqrt{\frac{2}{3} g} (y_{us} - y_w)^{0.5} \\ \frac{\partial f}{\partial y_{ds}} &= 0 \end{aligned} \quad (77)$$

• Untuk bendung arus menyelam:

$$\begin{aligned} f(y_{us}, y_{ds}) &= C_D B \sqrt{2 g} (y_{us} - y_{ds})^{0.5} (y_{ds} - y_w) \\ \frac{\partial f}{\partial y_{us}} &= C_D B \sqrt{0.5 g} (y_{us} - y_{ds})^{-0.5} (y_{ds} - y_w) \\ \frac{\partial f}{\partial y_{ds}} &= C_D B \sqrt{0.5 g} (y_{us} - y_{ds})^{-0.5} (2y_{us} - 3y_{ds} + y_w) \end{aligned} \quad (78)$$

### **Contoh 5: Bangunan pengambilan dengan pelimpah arus bebas (pintu Romijn)**

Persamaan 1: kontinuitas

$$\begin{aligned} Q_i^{n+1} - Q_{\text{pengambilan}}(t_{n+1}) &= Q_{i+1}^{n+1} \\ Q_i^n + \Delta Q_i - Q_{\text{pengambilan}}(t_{n+1}) &= Q_{i+1}^n + \Delta Q_{i+1} \\ Q_i^n + \Delta Q_i - Q_{\text{pengambilan}}(t_n) - \left. \frac{\partial f}{\partial y_{us}} \right]^n \Delta y_i &= Q_{i+1}^n + \Delta Q_{i+1} \end{aligned} \quad (79)$$

jika dibandingkan dengan Pers. (48), maka didapat:

$$A=0, \quad B=D=1, \quad C = -\left. \frac{\partial f}{\partial y_{us}} \right]^n, \quad G = Q_i^n - Q_{\text{pengambilan}}(t_n) - Q_{i+1}^n \quad (80)$$

Persamaan 2: Elevasi muka air dianggap sama antara titik i dan i+1, sehingga hasilnya seperti contoh sebelumnya:

$$\begin{aligned} y_i^{n+1} &= y_{i+1}^{n+1} \\ y_i^n + \Delta y_i &= y_{i+1}^n + \Delta y_{i+1} \end{aligned} \quad (81)$$

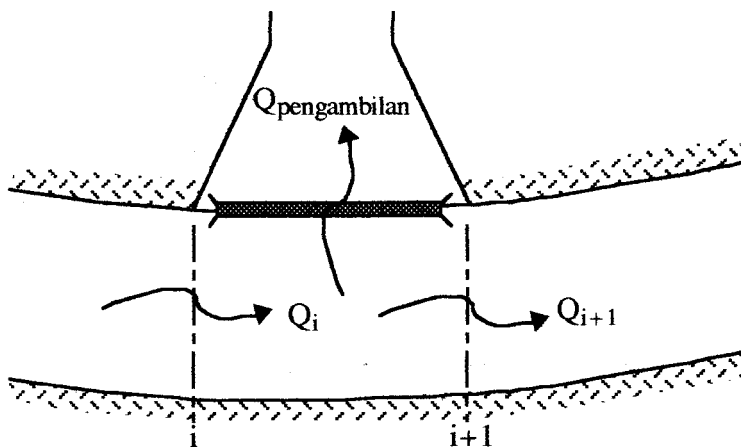
jika dibandingkan dengan Pers. (49), maka didapat:

$$AA=CC=1, \quad BB=DD=0, \quad GG = y_i^n - y_{i+1}^n \approx 0 \neq 0 \quad (82)$$

Sedangkan debit pengambilan dari pintu Romijn dapat dihitung dengan Pers. (69) yaitu persamaan untuk bendung arus bebas. Untuk lebih jelasnya maka debit pengambilan ditulis lagi sbb:

$$Q_{\text{pengambilan}} = C_D B \frac{2}{3} \sqrt{\frac{2}{3} g} (y_i - y_w)^{1.5} \quad (83)$$

dimana  $C_D$  adalah koefisien debit,  $B$  adalah lebar mercu Romijn,  $y_i$  adalah elevasi muka air sungai, dan  $y_w$  adalah elevasi mercu Romijn.



Gambar 10. Bangunan pengambilan arus bebas

## KESINGULERAN

Telah dijelaskan secara rinci di depan tentang hitungan aliran permanen untuk saluran tunggal dengan metoda 'sapuan-ganda.' Dalam bab ini dibahas mengenai timbulnya titik-titik singular dalam metoda tersebut. Yang dimaksud dengan kesinguleran adalah terdapatnya titik-titik hitungan yang menyebabkan suatu metoda numerik tidak dapat dilaksanakan secara tuntas. Hal itu disebabkan oleh adanya nilai-nilai suatu parameter yang mendekati tak terhingga atau adanya nilai penyebut dalam suatu persamaan yang mendekati nol.

Persamaan kerja metoda 'sapuan-ganda' adalah Pers. (23) dan (25) dengan koefisien pengaruh yang dinyatakan dalam Pers. (31) s/d (33). Didalam Pers. (33.c) s/d (33.e) untuk pias biasa dalam sungai alami pada umumnya nilai penyebut,  $C(DD)-(CC)D$ , tidak sama dengan nol, tetapi tidak demikian halnya dengan pias yang mempunyai kondisi batas internalnya.

Diambil contoh untuk bangunan bendung arus bebas, oleh karena suatu keadaan tertentu di hilir bendung, ditengah-tengah simulasi dapat terjadi bahwa arah alirannya membalik. Dari Pers. (68) didapat bahwa  $C=0$ , sedangkan dari Pers. (75) dan (77) diperoleh nilai  $CC=0$ , sehingga nilai penyebut,  $C(DD)-(CC)D$ , sama dengan nol.

Contoh yang lain: jika diinginkan suatu simulasi saluran dengan dilengkapi pintu sorong ataupun bendung, maka pemberian kondisi awal dengan elevasi muka air sama nilainya sepanjang saluran dapat menimbulkan masalah. Hal ini disebabkan karena didalam menghitung derivasi debit bendung ataupun pintu sorong terhadap elevasi ada faktor  $(y_{us}-y_{ds})^a$  didalam penyebut yang nilainya sama dengan nol, lihat Pers. (65), (66) dan (78).

Contoh yang lebih sering ditemui adalah pemberian kondisi awal berupa elevasi muka air yang sama nilainya sepanjang saluran dan nilainya lebih rendah dibandingkan elevasi mercu bendung. Hal ini menyebabkan tidak bisa dihitungnya debit bendung karena ada suku negatif yang dipangkatkan dengan bilangan pecahan. Selain itu memang fisiknya juga menggambarkan hal yang sama yaitu tidak ada aliran melalui mercu bendung.

Masalah-masalah di atas memang lebih banyak merupakan problem detail dalam teknik numerik, bukan merupakan problem pokok masalah aliran tak permanen. Untuk menangani masalah-masalah seperti di atas dengan sendirinya membutuhkan penanganan tersendiri yang setiap pakar numerik mempunyai penyelesaiannya sendiri-sendiri. Penyelesaian ini menjadi bagian penting dari suatu program komputer ditinjau dari segi teknik numerisnya, karena dapat menghindarkan program dari kemacetan non-prinsip. Oleh karena itu bagi pembaca yang tertarik diharapkan dapat membaca langsung program komputer yang disertakan, untuk melihat salah satu alternatif penyelesaian masalah-masalah di atas.

## CONTOH SOAL

1. Sebuah saluran beton sepanjang 24 km mempunyai kekasaran Strickler = 15 dan kemiringan dasar saluran 0.0005. Tampang saluran berbentuk segi empat dengan lebar dasar di bagian hulu 8 m dan berubah secara linier sehingga lebar dasar di bagian hilir menjadi 20 m. Dari sebelah hulu tercatat hidrograp debit masuk kedalam saluran sbb:

Waktu (menit)	Q (m <sup>3</sup> /detik)	Waktu (menit)	Q (m <sup>3</sup> /detik)
t < 0	100	75	300
0	100	90	250
15	100	105	200
30	100	120	150
45	250	135	100
60	350	t > 135	100

Dengan menggunakan panjang pias  $\Delta x = 1$  km,  $\beta = 0.5$ ,  $\alpha = 1.0$ ,  $\theta = 0.55$ , dan kondisi awal ( $t = 0$ ) sebagai berikut:

debit saluran,  $Q = 100 \text{ m}^3/\text{detik}$ ,

elevasi muka air,  $y(x) = 23.741 - 0.9653 x + 0.0097 x^2$

dimana satuan  $x$  adalah km dan satuan  $y$  adalah m. Pada hulu saluran  $x = 0$  km dan pada hilir saluran  $x = 24$  km. Di sebelah hilir, elevasi dasar saluran adalah 0.00 m. Waktu simulasi adalah 420 menit.

Catatan: persamaan muka air di atas adalah persamaan pendekatan dari profil muka air dengan debit tetap  $100 \text{ m}^3/\text{detik}$ ,

- a. Simulasikan aliran tak permanen dengan  $\Delta t = 15$  menit dan kondisi batas hilir adalah aliran seragam lokal  $Q = K\sqrt{S_0}$ , dimana  $K$  adalah faktor pengangkut saluran dan  $S_0$  adalah kemiringan dasar saluran. Hitunglah dan grafikkan hasil simulasi di beberapa tempat di sepanjang dan di hilir saluran!
- b. Dari hasil (a) akan didapat hidrograp elevasi muka air,  $y(t)$ , hidrograp debit,  $Q(t)$ , serta 'rating curve,'  $Q(y)$ , di bagian hilir saluran. Ulang simulasi di atas tiga kali dengan menggunakan kondisi batas hilir masing-masing  $y(t)$ ,  $Q(t)$ , dan  $Q(y)$  yang diperoleh dari hasil (a). [Catatan: jika program komputernya benar, maka hasil-hasil dari butir (b) akan selalu sama dengan hasil dari butir (a)].
- c. Ulangi simulasi pada butir (a) dengan  $\Delta t = 2, 5, 10, 15, 20$  menit dan 30 menit. Bandingkan hasilnya.
- d. Simulasikan aliran permanen dengan  $Q = 350 \text{ m}^3/\text{detik}$ , dengan cara membuat inflow hidrograp konstan dengan  $Q = 350 \text{ m}^3/\text{detik}$  dan bandingkan hasilnya dengan (a).
- e. Ulangi butir (a) dan tambahkan satu-persatu pada  $x = 14$  km, kondisi batas internal berupa:
  1. debit konstan  $50 \text{ m}^3/\text{detik}$  dari samping masuk kedalam saluran
  2. kantong banjir dengan elevasi dasar 10.75 m dan luas tampungan konstan sebesar  $50.000 \text{ m}^2$
  3. pintu sorong dengan koefisien debit 0.68 dan luas bukaannya  $75 \text{ m}^2$
  4. bendung rendah dengan koefisien debit 1.0, lebar mercu 15.0 m, elevasi mercu 10.0 m, dan tinggi bendung 6.0 m
  5. bendung rendah, sama dengan butir (4), ditambah dengan pintu Romijn sebagai bangunan pengambilan pengairan yang diletakkan 1 km disebelah hulu bendung. Untuk pintu Romijn, koefisien debitnya 1.0, lebar mercu 10.0 m, dan elevasi mercunya 14.0 m.
- f. Ulangi butir (a) untuk nilai kekasaran saluran, koefisien Strickler, berkisar dari 10 s/d 30. Catatan: hubungan antara koefisien Manning dan Strickler adalah  $n_M * k_S = 1$ .

Cobalah tampilkan dan diskusikan hasilnya semenarik mungkin. Tariklah kesimpulan yang berguna untuk praktek di lapangan.

## CARA PENYELESAIAN

Untuk mensimulasikan contoh soal di atas, maka dalam buku ini disertakan komputer program yang bernama ATPIDUST dengan bahasa FORTRAN yang bersifat umum. Untuk menangani bentuk geometri saluran yang tidak teratur, hanya dibutuhkan modifikasi pada satu subroutine yang bernama GEOMETRY. Jenis kondisi batas internal dalam program tersebut ada lima macam dan masing-masing hanya satu buah. Untuk menambah macam bangunan maka ATPIDUST harus ditambah dengan subroutine baru untuk menangani bangunan tersebut. Cara penjabaran rumus dari bangunan ybs sejalan dengan penjelasan yang diberikan dalam bab 'KONDISI BATAS INTERNAL.' Untuk menambah bangunan sejenis tetapi lebih dari satu, hanya dibutuhkan perubahan pada stuktur datanya dari satu simpel variabel menjadi array variabel. Program komputer ATPIDUST dibuat umum agar pemakai dapat mencoba berbagai macam masalah saluran tunggal dengan leluasa.

Program komputer ATPIDUST ini dapat dipakai untuk menyelesaikan masalah di lapangan. Untuk itu pemakai program dituntut untuk mencocokkan kondisi lapangan dengan kebutuhan data dari program komputer ini. Untuk keperluan rancangan maupun untuk simulasi keadaan yang akan datang, dibutuhkan kalibrasi parameter yang ada dalam program dengan data di lapangan dimana simulasi tersebut akan dilaksanakan.

Program komputer ini dapat pula dipakai oleh para pengajar maupun mahasiswa yang sedang mengajar atau belajar hidrolika numerik untuk menyampaikan/mendalami konsep-konsep tertentu misalnya pengaruh skema implisit ( $\theta=1$ ) dan eksplisit ( $\theta=0$ ) terhadap hasil simulasi. Demikian pula halnya dengan pengaruh bilangan Courant terhadap hasil simulasi, dimana bilangan Courant didefinisikan sebagai

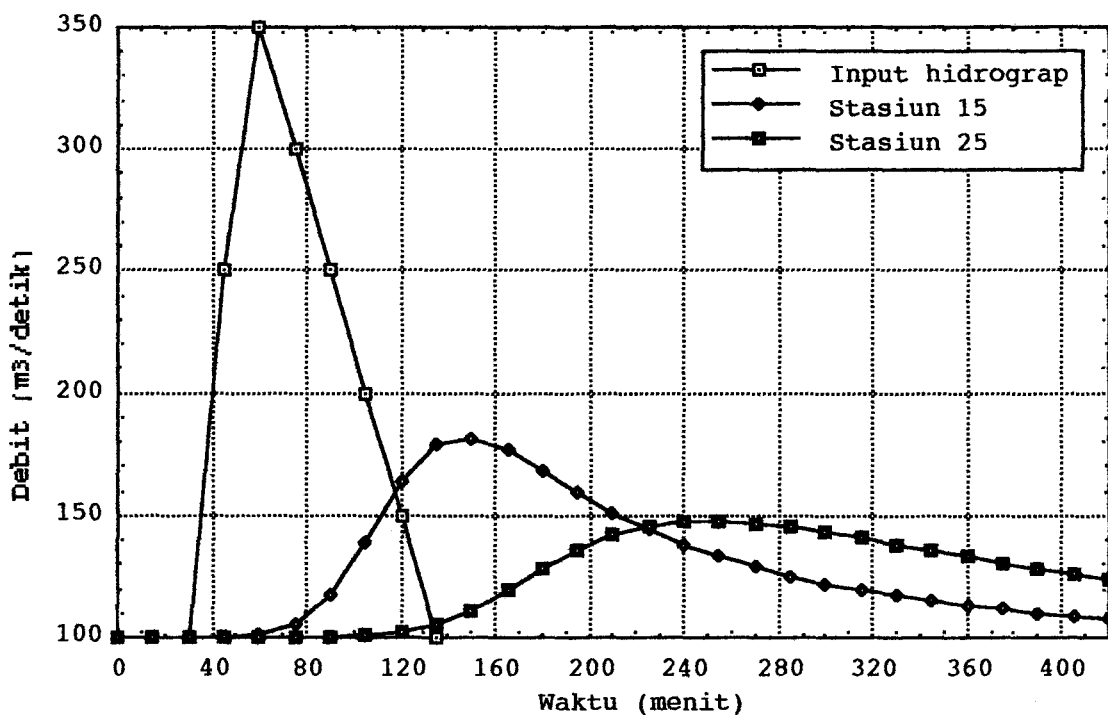
$$Cr = \left| V + \sqrt{gh_0} \right| \frac{\Delta t}{\Delta x} \quad (84)$$

dimana Cr adalah bilangan Courant, V adalah kecepatan rerata aliran, dan  $h_0$  adalah dalam rerata air yang dihitung sebagai luas tampang basah dibagi lebar muka air.

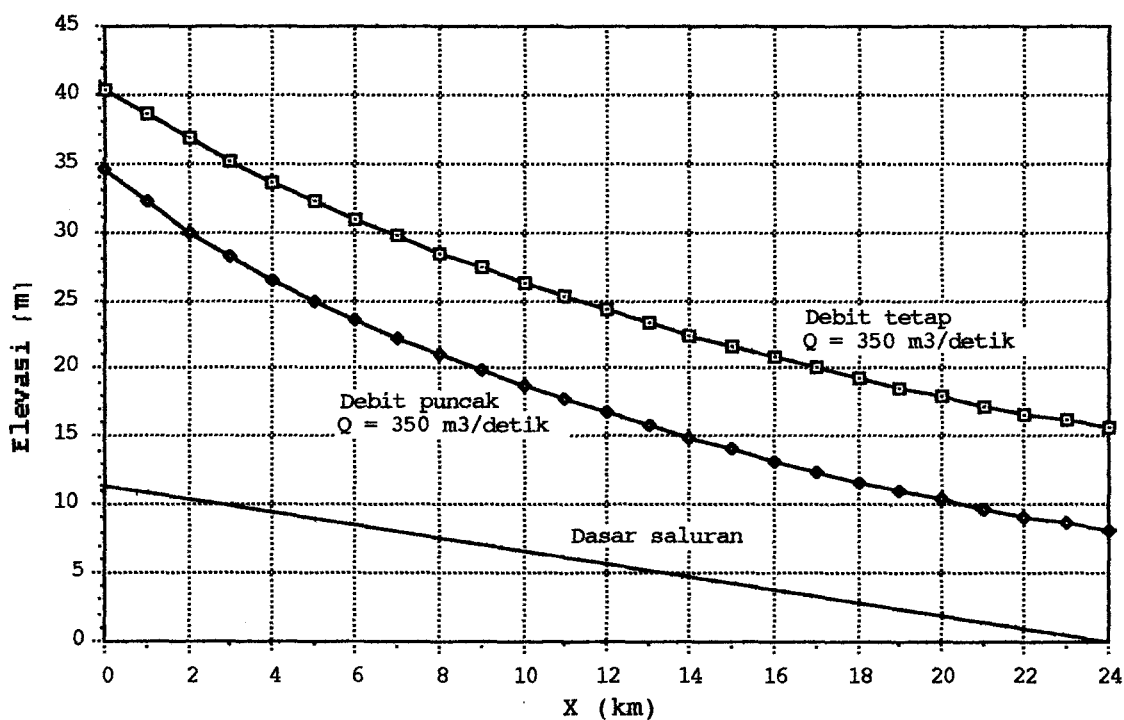
## BEBERAPA HASIL SIMULASI

Beberapa hasil simulasi dari contoh soal di atas, yang diselesaikan dengan komputer program ATPIDUST tersebut di atas, disajikan dan dibahas secara singkat dalam bab ini.

Hasil dari soal pada butir (a) disajikan dalam Gambar 11, dimana input hidrograp dengan debit puncak  $350 \text{ m}^3/\text{detik}$  mengalami peredaman debit sepanjang saluran. Pada stasiun 15, yang terletak 14 km sebelah hilir, debit puncak menjadi  $182 \text{ m}^3/\text{detik}$ . Pada stasiun 25, yang terletak 25 km sebelah hilir, debit puncak nilainya menjadi  $148 \text{ m}^3/\text{detik}$  dan tiba 195 menit ( $= 3.25 \text{ jam}$ ) setelah debit puncak banjir datang disebelah hulu saluran.



Gambar 11. Hidrograp debit pada stasiun 15 dan 25 dengan  $k_s = 15$



Gambar 12. Elevasi muka air maksimum sepanjang saluran dengan  $k_s = 15$

Dalam praktek dilapangan, sebagai suatu contoh, jika input debit merupakan hidrograp banjir yang terjadi di sebelah hulu, maka hasil hitungan dapat dipakai sebagai data pendukung pada sistim pengendalian maupun peringatan banjir yang disampaikan kepada penduduk di sepanjang sungai. Jika input hidrograp merupakan ketersediaan debit pengambilan untuk suatu saluran primer, maka hasil simulasi dapat dipakai untuk memperkirakan jumlah air yang tersedia di sepanjang saluran primer.

Pada Gambar 12 disajikan elevasi muka air maksimum yang terjadi selama simulasi di setiap titik-titik hitungan sepanjang saluran, baik yang dihitung dengan aliran permanen maupun tak permanen.

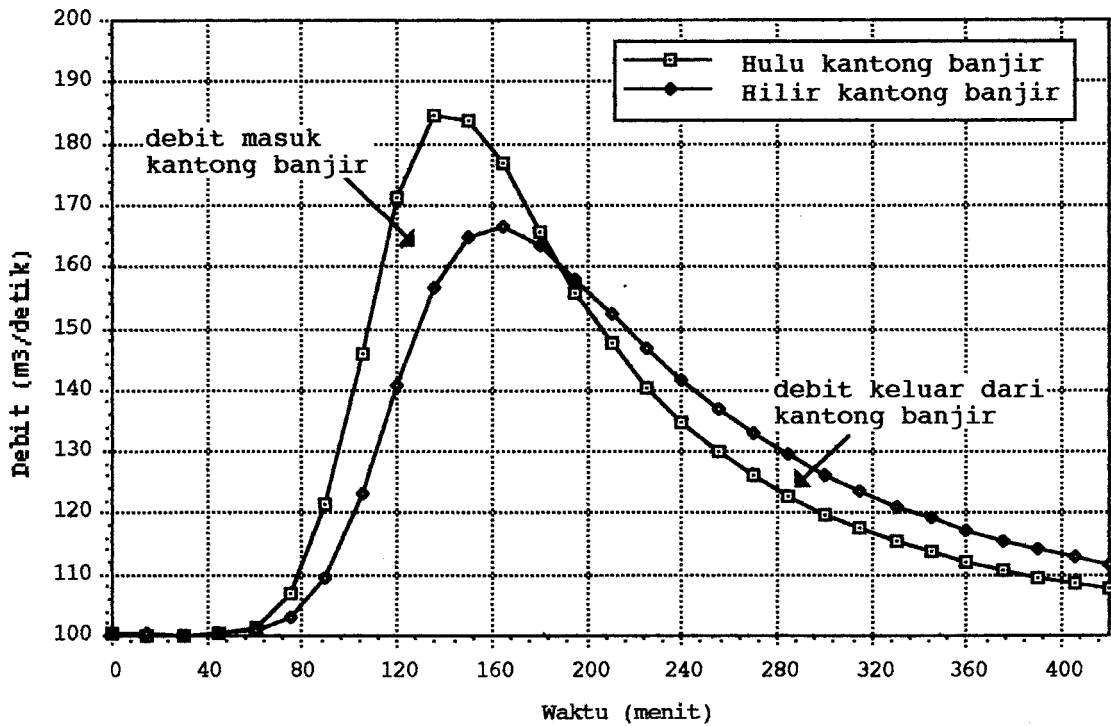
Misalkan ada sungai akan diberi tanggul, maka informasi yang didapat dari hasil hitungan ini sangat berguna untuk menentukan elevasi puncak tanggul. Hasil hitungan dalam Gambar 12 menunjukkan bahwa dengan menggunakan aliran tak permanen, elevasi puncak tanggul lebih rendah dibanding dengan hitungan menggunakan aliran permanen, sehingga terdapat penghematan dalam pembuatan tanggul. Untuk nilai  $k_s = 15$  selisih elevasi muka air maksimum dapat mencapai 5 m. Tentu saja pada praktek di lapangan perlu pula dipertimbangkan penentuan nilai  $k_s$  yang cocok untuk sungai atau saluran yang menjadi obyek simulasi. Oleh karena itu di bawah juga disajikan hasil simulasi untuk berbagai nilai  $k_s$ .

Untuk mensimulasi pengaruh adanya kantong banjir pada suatu sungai, maka pada jarak 14 km dari hulu dianggap ada kantong banjir yang mempunyai data seperti telah disebut dalam contoh soal. Hasil simulasi disajikan dalam Gambar 13, dimana debit sebelah hulu dan hilir dari kantong banjir diplot bersama. Terlihat bahwa kantong banjir yang tersedia dapat mengurangi debit puncak di daerah sekitarnya sekitar  $18 \text{ m}^3/\text{detik}$  (atau sebesar 10% dari debit puncak setempat), dengan waktu puncak menjadi mundur selama 30 menit. Tentu saja reduksi debit puncak maupun pengunduran waktu puncak sangat dipengaruhi oleh ukuran dari kantong banjir.

Tampak dari Gambar 13, bahwa program ATP1DUST yang dipakai benar-benar mensimulasi sifat fisik dari kantong banjir yaitu mengurangi debit sungai pada waktu puncak banjir lewat dan mengosongkan isi kantong pada waktu puncak banjir telah lewat. Kemampuan mensimulasi kantong banjir ini sangat berguna didalam praktek terutama untuk menanggulangi masalah banjir.

Contoh: misalkan pada suatu aliran sungai yang sering banjir terdapat suatu daerah di tepian sungai yang mempunyai potensi untuk dikembangkan menjadi kantong banjir. Untuk mengetahui bagaimana penyiapan kantong banjir tersebut mungkin dibutuhkan informasi mengenai seberapa besar reduksi debit puncak yang dihasilkan. Jika reduksi tersebut masih belum mencukupi apakah masih mungkin memperluas kantong banjir. Berapa besar luas dan berapa dalam dasar kantong banjir harus dibuat agar sesuai dengan rencana penanggulangan banjir sungai tersebut. Hal-hal seperti di atas dapat dijawab dengan mensimulasikan dengan program komputer yang cukup sederhana ini.



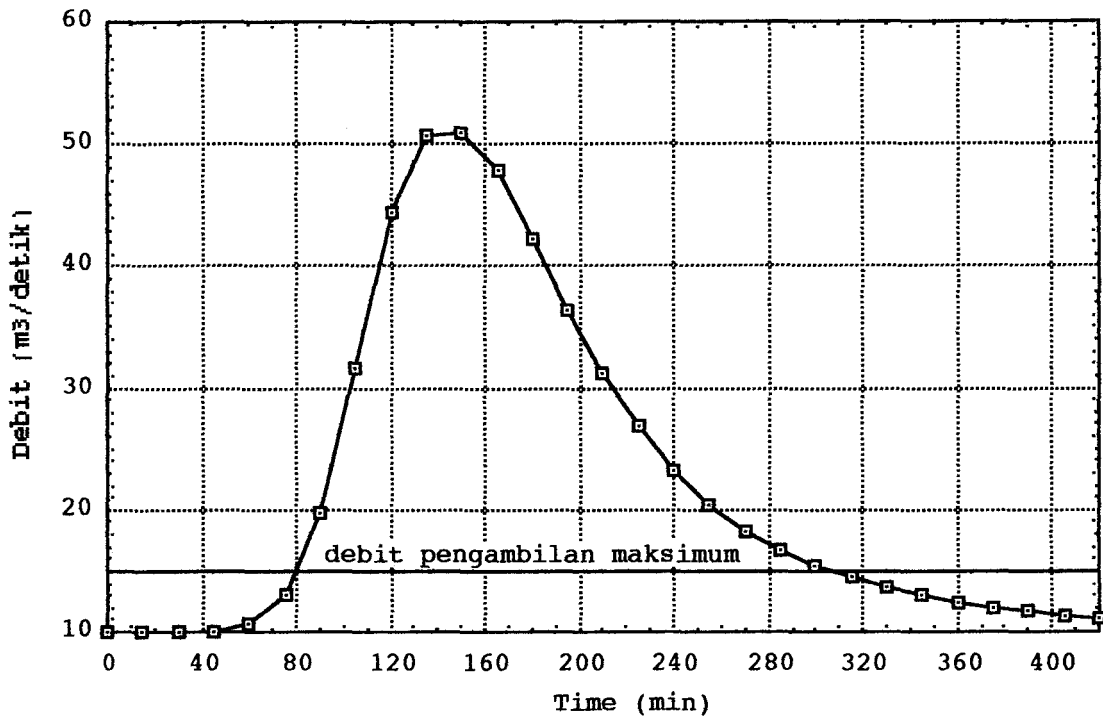


Gambar 13. Hidrograp debit di sebelah hulu dan hilir kantong banjir

Pada Gambar 14 di bawah ini disajikan grafik debit pengambilan untuk pengairan dengan pintu Romijn, pada waktu terjadi banjir. Pintu pengambilan Romijn dalam simulasi ini diletakkan pada jarak 1 km sebelah hulu bendung. Bendungnya sendiri merupakan bendung rendah yang lokasinya 14 km dari hulu sungai tempat banjir datang.

Pada kondisi normal dengan debit sungai konstan sebesar  $100 \text{ m}^3/\text{detik}$ , bangunan pengambilan beroperasi secara normal pula dengan debit pengambilan sebesar  $10 \text{ m}^3/\text{detik}$  (lihat Gambar 14). Saat debit banjir sebesar  $350 \text{ m}^3/\text{detik}$  tiba di hulu sungai pada waktu  $t = 60$  menit, pada pintu Romijn pengaruh banjir sudah terasa, dengan adanya kenaikan sedikit pada debit pengambilannya. Dari waktu tersebut, debit pengambilan akan semakin membesar seperti yang terlihat pada Gambar 14.

Jika dikehendaki, pada saluran pengambilan, debit pengambilan maksimum diperkenankan hanya  $15 \text{ m}^3/\text{detik}$ , maka pada waktu  $t = 80$  s/d 320 menit pintu pengambilan Romijn harus ditutup. Jadi dari hasil simulasi dapat diperkirakan akan terjadi penutupan pintu pengambilan Romijn selama  $300 - 80 = 220$  menit atau 3.67 jam. Jika waktu empat jam tersebut diperkirakan terlalu lama, mungkin dapat diadakan simulasi lagi dengan menaikkan elevasi mercu pintu pengambilan Romijn selama terjadi banjir, untuk melihat pengaruh kenaikan tersebut pada pengoperasian pintu Romijn di waktu terjadi banjir.



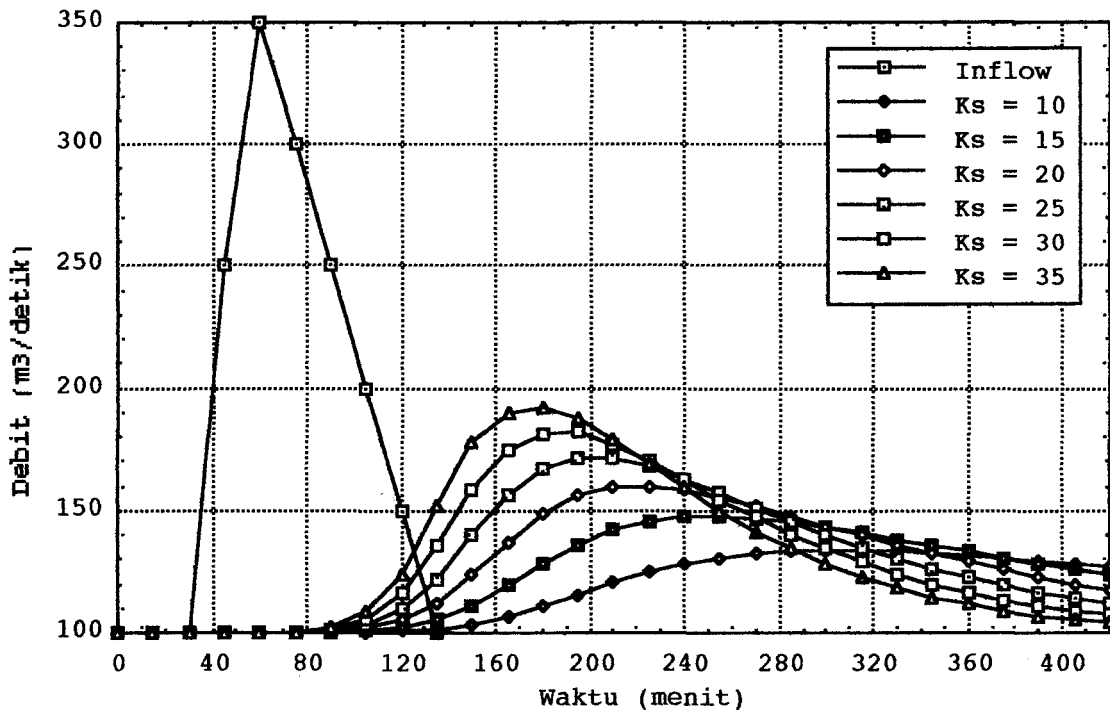
Gambar 14. Debit pada bangunan pengambilan Romijn

Aspek penting yang harus diperhatikan dalam hitungan aliran tak permanen dalam sungai/saluran adalah penentuan nilai koefisien Manning ataupun Strickler. Pengaruh nilai kekasaran sungai/saluran dapat juga disimulasikan dengan program ATP1DUST ini. Dalam Gambar 15 dan 16 disajikan hasil simulasi dengan menggunakan nilai kekasaran Strickler,  $k_s$ , yang berkisar dari 10 s/d 35.

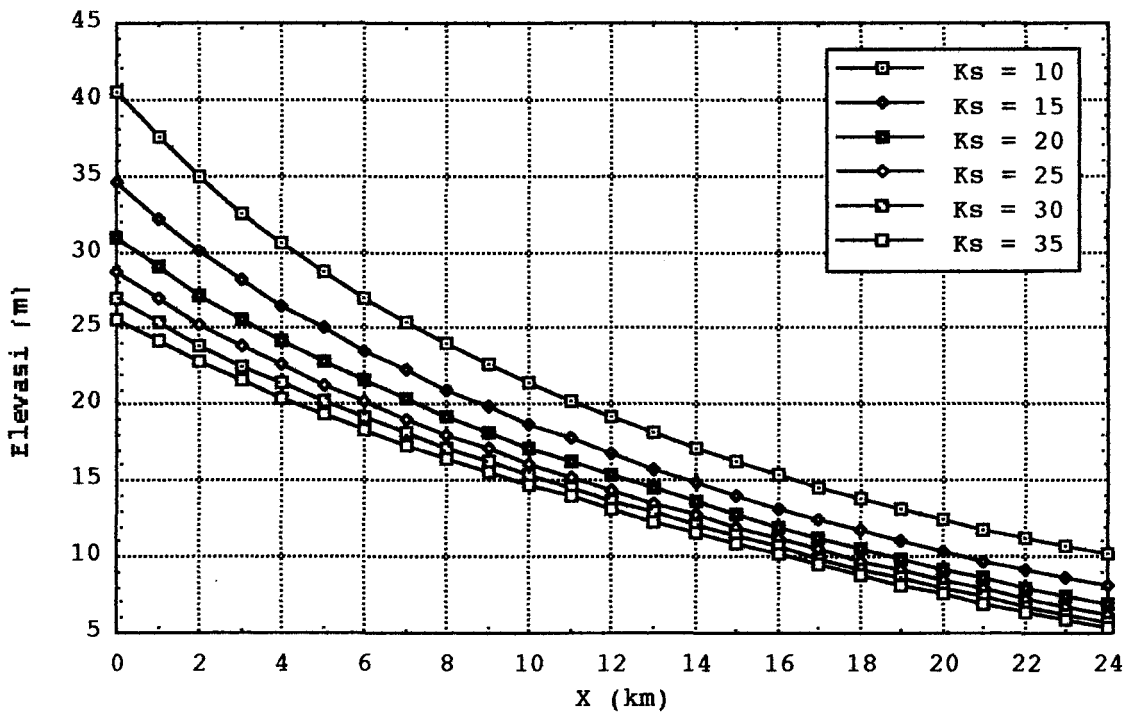
Dalam Gambar 15 disajikan hidrograp debit untuk nilai  $k_s$  seperti diberikan dalam contoh soal di atas. Tampak bahwa makin tinggi nilai  $k_s$  makin besar debit puncaknya dan makin cepat datangnya. Untuk nilai  $k_s$  yang paling tinggi sekalipun (dalam contoh adalah 35), debit puncaknya masih dibawah  $200 \text{ m}^3/\text{detik}$ .

Dalam Gambar 16 disajikan elevasi muka air maksimum untuk nilai  $k_s$  sama dengan yang digunakan dalam simulasi sebelumnya. Dari hasil yang disajikan menunjukkan bahwa pada nilai  $k_s$  terendah (= 10) mengakibatkan elevasi muka air maksimum sepanjang saluran mempunyai nilai tertinggi.

Dari kedua hasil di bawah didapat suatu gambaran mekanisme pengaruh dari nilai kekasaran Strickler,  $k_s$ . Semakin rendah nilai  $k_s$ , saluran semakin mampu menyediakan tampungan sementara untuk debit besar, sehingga elevasi muka air lebih tinggi tetapi kecepatan datang debit puncak lebih lambat dan lebih kecil. Tampak pula dari kedua hasil di bawah bahwa korelasi antara nilai  $k_s$  dengan debit puncak, waktu puncak dan elevasi muka air maksimum tidak linier. Bagi yang tertarik mungkin dapat membuat grafik korelasi tersebut sehingga didapat gambaran yang lebih jelas mengenai korelasi tersebut. Korelasi semacam ini sangat dibutuhkan perencana untuk mendapatkan gambaran yang jelas bagaimana pengaruh penentuan nilai  $k_s$  terhadap saluran yang akan dirancang.



Gambar 15. Hidrograp debit di sebelah hilir saluran untuk beberapa nilai  $k_s$



Gambar 16. Elevasi muka air saluran maksimum, untuk beberapa nilai  $k_s$

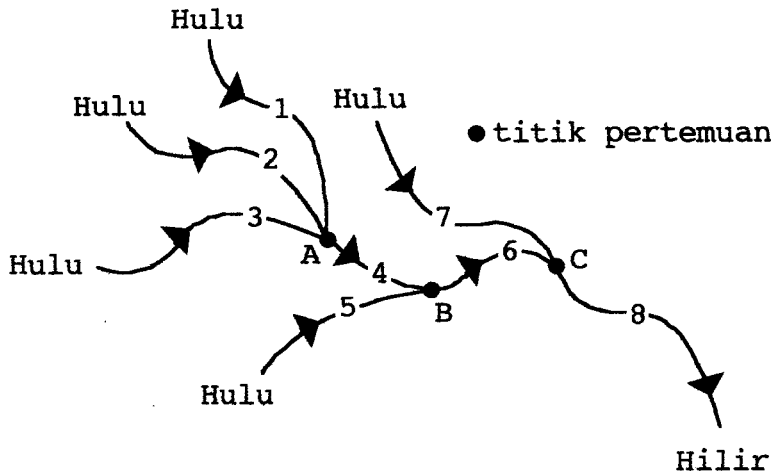
Dari contoh soal dan pembahasan beberapa hasil yang relatif sederhana dan mudah dipahami, diharapkan merangsang kesadaran pembaca akan potensi hitungan aliran tak permanen pada bidang keairan. Program ATP1DUST yang dipakai untuk menyelesaikan contoh soal di atas, cukup sederhana dan mudah dikembangkan, sehingga dapat diubah untuk keperluan khusus yang dihadapi oleh pembaca. Di bawah ini dijelaskan bagaimana cara mengembangkan program tersebut untuk menangani jaringan anak sungai yang saling bertemu dan membentuk sungai yang lebih besar.

## PERTEMUAN BEBERAPA SALURAN

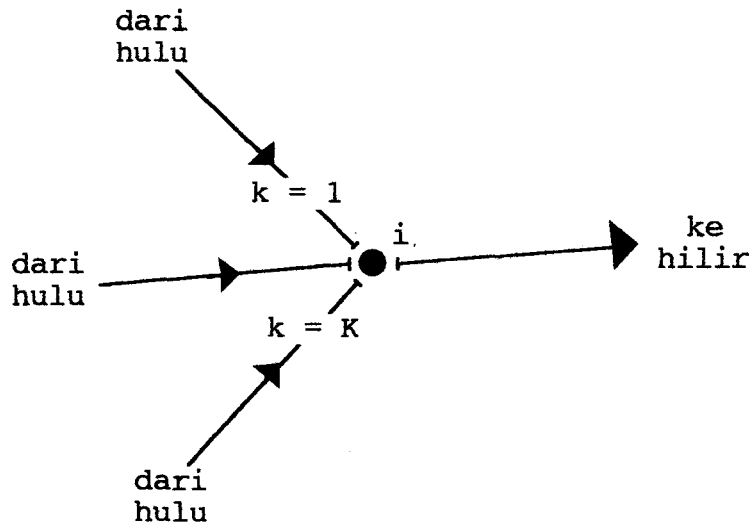
Metoda 'sapuan-ganda' yang diterangkan secara rinci di depan, dapat langsung diterapkan pada anak-anak sungai yang bertemu membentuk suatu sungai yang lebih besar. Yang dimaksud dengan titik pertemuan pada topik ini secara teknis dapat didefinisikan sebagai suatu titik dimana terdapat beberapa saluran/sungai yang bermuara di titik tersebut, tetapi hanya terdapat satu saluran/sungai yang berasal dari titik tersebut. Untuk lebih jelasnya dapat dilihat dalam Gambar 17.

Penerapan langsung dari metoda 'sapuan-ganda' saluran tunggal untuk sistim sungai/saluran pada Gambar 17.a adalah sebagai berikut:

1. 'sapuan ke hilir' selalu dimulai dari hulu ke hilir dengan menghitung koefisien pengaruh E, F, L, M, dan N. Jadi 'sapuan ke hilir' dapat dimulai dari anak sungai nomer 1, 2 dan 3, dengan menggunakan kondisi batas hulu masing-masing untuk menghitung  $E_1$  dan  $F_1$ . Sampai di cabang A dengan menggunakan Pers. (88) dan (89) koefisien pengaruh untuk anak sungai nomer 4 dapat dihitung. Sebelum hitungan dilanjutkan ke anak sungai nomer 6, koefisien pengaruh dari anak sungai nomer 4 harus dihitung terlebih dahulu. Setelah anak sungai nomer 4 dan 5 terhitung koefisien pengaruhnya, maka hitungan berlanjut ke anak sungai nomor 6 dan 7, untuk kemudian berakhir pada induk sungai nomer 8.
2. 'sapuan ke hulu' dimulai dari titik hitungan paling hilir menuju ke hulu dengan menghitung koreksi  $\Delta y$  dan  $\Delta Q$  pada masing-masing titik-titik hitungan, di setiap anak sungai. Jadi  $\Delta y$  dan  $\Delta Q$  dihitung pertama kali untuk sungai nomer 8 dengan menggunakan kondisi batas hilir. Hitungan dilanjutkan sampai titik pertemuan C dimana  $\Delta y_C$  telah diketahui dari hitungan pada sungai nomer 8. Koreksi elevasi muka air untuk semua anak sungai yang bermuara di titik C dihitung dengan Pers. (86). Jadi  $\Delta y_k$  dari Pers. (86) dan  $\Delta Q_k$  dari Pers. (84) inilah yang digunakan untuk 'sapuan ke hulu' selanjutnya pada anak sungai nomer 6 dan 7. Demikian selanjutnya hitungan dilanjutkan sampai semua elevasi muka air maupun debit di masing-masing anak sungai selesai dihitung.



(a) Sistem sungai yang cocok untuk penerapan langsung metoda 'sapan-ganda' untuk satu saluran



(b) Skema penerapan metoda 'sapan-ganda' pada titik-titik pertemuan saluran/sungai

Gambar 17. Sistem pertemuan sungai dan skema cara hitungannya

3. Cara hitungan koefisien pengaruh di titik pertemuan dijelaskan secara rinci di bawah ini (lihat Gambar 17.b). Dari 'sapan ke hilir' pada masing-masing anak sungai yang bermuara di titik pertemuan, untuk titik hitungan paling hilir, dapat ditulis hubungan sbb:

$$\Delta Q_k = E_k \Delta y_k + F_k \quad \text{untuk } k = 1, \dots, K \quad (84)$$

Pada titik-titik pertemuan hukum kontinuitas masih tetap berlaku dan dianggap bahwa elevasi muka air di titik pertemuan adalah sama. Jadi pada titik pertemuan berlaku hubungan

$$\begin{aligned}
Q_i^{n+1} &= \sum_{k=1}^{k=K} Q_k^{n+1} \\
Q_i^n + \Delta Q_i &= \sum_{k=1}^{k=K} [Q_k^n + \Delta Q_k] \\
&= \sum_{k=1}^{k=K} [Q_k^n + E_k \Delta y_k + F_k] \\
\Delta Q_i &= \sum_{k=1}^{k=K} [Q_k^n + E_k \Delta y_k + F_k] - Q_i^n
\end{aligned} \tag{85}$$

Jika elevasi muka air di titik pertemuan dianggap sama, maka

$$\begin{aligned}
y_i^{n+1} &= y_k^{n+1} \\
y_i^n + \Delta y_i &= y_k^n + \Delta y_k \\
\Delta y_k &= y_i^n + \Delta y_i - y_k^n
\end{aligned} \tag{86}$$

Substitusi Pers. (86) kedalam Pers. (85) menghasilkan

$$\begin{aligned}
\Delta Q_i &= \sum_{k=1}^{k=K} [Q_k^n + E_k (y_i^n + \Delta y_i - y_k^n) + F_k] - Q_i^n \\
&= \sum_{k=1}^{k=K} [Q_k^n + E_k (y_i^n + \Delta y_i - y_k^n) + F_k] - Q_i^n \\
\text{atau} \\
\Delta Q_i &= \left[ \sum_{k=1}^{k=K} E_k \right] \Delta y_i + \sum_{k=1}^{k=K} Q_k^n - Q_i^n + \sum_{k=1}^{k=K} [E_k (y_i^n - y_k^n) + F_k]
\end{aligned} \tag{87}$$

Jika Pers. (87) dibandingkan dengan Pers. (32) yaitu  $\Delta Q_i = E_i \Delta y_i + F_i$ , maka diperoleh nilai  $E_i$  dan  $F_i$  sbb:

$$E_i = \sum_{k=1}^{k=K} E_k \tag{88}$$

$$F_i = \sum_{k=1}^{k=K} Q_k^n - Q_i^n + \sum_{k=1}^{k=K} [E_k (y_i^n - y_k^n) + F_k] \tag{89}$$

4. Jika pada salah satu atau beberapa anak sungai terdapat bangunan, maka penyelesaiannya sama dengan penanganan kondisi batas internal pada saluran tunggal yang telah dijelaskan secara rinci di depan.

Ditinjau dari teknik numerisnya, hitungan untuk saluran/sungai tunggal dibandingkan dengan hitungan untuk saluran/sungai dengan titik-titik pertemuan tidak berbeda banyak. Dari segi pemrograman, saluran/sungai dengan titik-titik pertemuan membutuhkan manajemen

data yang lebih canggih, karena urutan penggunaan 'sapan ke hilir and ke hulu' sudah tertentu untuk suatu sistim saluran/sungai. Sehingga sebelum melangkah ke pemrograman hitungan untuk sungai/saluran dengan titik-titik pertemuan, pembaca disarankan untuk membaca dan memahami betul teknik numeris serta pemrogramannya untuk saluran/sungai tunggal.

## CONTOH MASUKAN DAN KELUARAN

Di bawah ini disajikan dua buah contoh masukan yang dibutuhkan oleh program komputer ATPIDUST. Hitungan yang dihasilkan disajikan pula setelah masukan. Tidak semua masukan dan keluaran yang dibahas di depan disertakan dalam buku ini. Dua masukan dan keluaran in hanya sebagai contoh; satu masukan tanpa kondisi batas internal dan yang kedua dengan kondisi batas internal. Kemudian disajikan pula struktur dari data masukan, sehingga pembaca dapat memasukkan data sesuai dengan yang dibutuhkan oleh programnya.

### FILE DATA MASUKAN 1:

```

3
1-D unsteady flow in rectangular channel.
This run will provide data for the next
several runs.
9.81 1.0 0.5 0.55 420.0 15.0 15
25
T
1000.0
15.0 0.0005
3
1 2 15 25
1
20
  0.0 100.0
 15.0 100.0
 30.0 100.0
 45.0 250.0
 60.0 350.0
 75.0 300.0
 90.0 250.0
105.0 200.0
120.0 150.0
135.0 100.0
3
0

```

### FILE HASIL HITUNGAN 2:

```

=====
THE RESULTS OF SIMULATION OF 1-D UNSTEADY FLOW IN SINGLE CHANNEL
=====
TITLE OF THE RUN:

```

1-D unsteady flow in rectangular channel.  
This run will provide data for the next  
several runs.

## INPUT FILE:

input

## OUTPUT FILE:

output1

## GENERAL DATA:

Gravity acceleration = 9.81 m/sec<sup>2</sup>  
Alpha = 1.00  
Beta = 0.50  
Theta = 0.55  
Simulation Time = 420.00 minutes  
Time step = 15.00 minutes  
Number of time step = 28  
Number of points = 25  
Steady state run = 15 time steps

## MAIN CHANNEL:

Length = 24.00 km  
K Strikler = 15.00  
Bottom Slope = 0.50E-03

## BOUNDARY CONDITION:

U/S: 1.  $Q(t)$ , discharge given as a function of time  
D/S: 3.  $Q(y) = \text{Conveyance} \cdot \sqrt{S_0}$ , locally uniform flow

## NO INTERNAL BOUNDARY CONDITION

## OUTPUT CONTROLS:

Frequency of printout = 1 time steps  
Number of Location = 2  
At Location = 15 25

## ELEVATION AND DISCHARGE HYDROGRAPHS

Minute	Y( 15)	Q( 15)	Y( 25)	Q( 25)
-----				
Steady state stabilization:				
-225.0	12.13	100.00	6.16	100.00
-210.0	12.16	101.02	6.14	100.38
-195.0	12.18	101.03	6.12	99.80
-180.0	12.19	100.97	6.12	99.81
-165.0	12.19	100.71	6.12	99.81
-150.0	12.19	100.63	6.12	99.95
-135.0	12.19	100.52	6.13	100.05
-120.0	12.18	100.45	6.13	100.16
-105.0	12.18	100.38	6.13	100.22
-90.0	12.18	100.33	6.14	100.28
-75.0	12.18	100.28	6.14	100.31
-60.0	12.18	100.24	6.14	100.32
-45.0	12.18	100.21	6.14	100.33
-30.0	12.17	100.18	6.14	100.32
-15.0	12.17	100.16	6.14	100.31
Unsteady computation begin:				
0.0	12.17	100.13	6.14	100.30
15.0	12.17	100.12	6.14	100.28
30.0	12.17	100.10	6.14	100.26
45.0	12.17	100.18	6.13	100.25
60.0	12.19	101.10	6.13	100.23
75.0	12.27	105.38	6.13	100.24



90.0	12.54	117.32	6.14	100.36
105.0	13.07	139.09	6.16	100.85
120.0	13.78	163.99	6.22	102.29
135.0	14.39	178.92	6.36	105.49
150.0	14.75	181.50	6.60	111.15
165.0	14.90	177.18	6.94	119.12
180.0	14.87	168.35	7.31	128.00
195.0	14.74	159.37	7.63	135.94
210.0	14.58	151.32	7.87	141.85
225.0	14.41	144.22	8.02	145.53
240.0	14.23	138.24	8.09	147.27
255.0	14.06	133.08	8.10	147.52
270.0	13.90	128.76	8.07	146.71
285.0	13.75	125.05	8.00	145.15
300.0	13.61	121.91	7.92	143.11
315.0	13.47	119.21	7.83	140.78
330.0	13.35	116.87	7.73	138.29
345.0	13.24	114.86	7.62	135.73
360.0	13.13	113.10	7.52	133.18
375.0	13.04	111.56	7.42	130.69
390.0	12.95	110.22	7.32	128.28
405.0	12.87	109.04	7.22	125.98
420.0	12.80	108.00	7.13	123.81

## ELEVATION AND DISCHARGE ALONG THE CHANNEL

i	X(m)	YMAX(m)	Minute	Y(m)	Q(cms)	Minute
1	0.00	34.54	75.00	24.06	100.00	420.00
2	1000.00	32.18	75.00	22.99	100.16	420.00
3	2000.00	30.03	90.00	21.99	100.36	420.00
4	3000.00	28.23	90.00	21.04	100.60	420.00
5	4000.00	26.49	105.00	20.14	100.90	420.00
6	5000.00	25.00	105.00	19.28	101.27	420.00
7	6000.00	23.53	120.00	18.46	101.70	420.00
8	7000.00	22.26	120.00	17.67	102.19	420.00
9	8000.00	21.00	120.00	16.91	102.76	420.00
10	9000.00	19.87	135.00	16.17	103.42	420.00
11	10000.00	18.76	135.00	15.46	104.16	420.00
12	11000.00	17.73	150.00	14.77	104.98	420.00
13	12000.00	16.74	150.00	14.10	105.90	420.00
14	13000.00	15.80	165.00	13.44	106.90	420.00
15	14000.00	14.90	165.00	12.80	108.00	420.00
16	15000.00	14.04	180.00	12.18	109.19	420.00
17	16000.00	13.20	180.00	11.57	110.47	420.00
18	17000.00	12.42	195.00	10.97	111.85	420.00
19	18000.00	11.68	210.00	10.38	113.31	420.00
20	19000.00	10.99	225.00	9.81	114.86	420.00
21	20000.00	10.34	240.00	9.25	116.50	420.00
22	21000.00	9.73	240.00	8.70	118.22	420.00
23	22000.00	9.15	240.00	8.16	120.02	420.00
24	23000.00	8.61	255.00	7.64	121.88	420.00
25	24000.00	8.10	255.00	7.13	123.81	420.00

FILE DATA MASUKAN 2:

```

2
This run used the data from the original run
plus weir at i = 15 and intake at i = 14
9.81 1.0 0.5 0.55 420.0 15.0 30
25
T
1000.0
15.0 0.0005
3
1 1 14
1
20
  0.0 100.0
 15.0 100.0
 30.0 100.0
 45.0 250.0
 60.0 350.0
 75.0 300.0
 90.0 250.0
105.0 200.0
120.0 150.0
135.0 100.0
3
2
15 4
1.0 15.0 10.0 6.0
14 5
1.0 10.0 14.0

```

FILE HASIL HITUNGAN 2:

```

=====
THE RESULTS OF SIMULATION OF 1-D UNSTEADY FLOW IN SINGLE CHANNEL
=====

```

## TITLE OF THE RUN:

This run used the data from the original run  
plus weir at i = 15 and intake at i = 14

## INPUT FILE:

input

## OUTPUT FILE:

out\_weir\_intake

## GENERAL DATA:

```

Gravity acceleration = 9.81 m/sec2
Alpha                = 1.00
Beta                 = 0.50
Theta                = 0.55
Simulation Time      = 420.00 minutes
Time step            = 15.00 minutes
Number of time step = 28
Number of points     = 27
Steady state run     = 30 time steps

```

## MAIN CHANNEL:

```

Length              = 24.00 km
K Strikler          = 15.00
Bottom Slope        = 0.50E-03

```

## BOUNDARY CONDITION:

U/S: 1.  $Q(t)$ , discharge given as a function of time

D/S: 3.  $Q(y) = \text{Conveyance} \cdot \sqrt{S_0}$ , locally uniform flow

INTERNAL BOUNDARY CONDITION:

4: Weir at  $x = 14.0$  km ( $i = 16$ )

Disch.Coeff. = 1.00

Crest Width = 15.00 m

Crest Elev. = 10.00 m

Height = 6.00 m

5: Free flowing intake at  $x = 13.0$  km ( $i = 14$ )

Disch.Coeff. = 1.00

Crest Width = 10.00 m

Crest Elev. = 14.00 m

OUTPUT CONTROLS:

Frequency of printout = 1 time steps

Number of Location = 2

At Location = 14 15

ELEVATION AND DISCHARGE HYDROGRAPHS

Minute	Y( 14)	Q( 14)	Y( 15)	Q( 15)
-----				
Steady state stabilization:				
-450.0	12.83	100.00	12.83	100.00
-435.0	12.88	101.28	12.88	101.28
-420.0	12.97	98.54	12.97	98.54
-405.0	13.07	97.45	13.07	97.45
-390.0	13.12	97.95	13.12	97.95
-375.0	13.15	98.66	13.15	98.66
-360.0	13.16	98.99	13.16	98.99
-345.0	13.17	99.20	13.17	99.20
-330.0	13.18	99.36	13.18	99.36
-315.0	13.18	99.47	13.18	99.47
-300.0	13.19	99.56	13.19	99.56
-285.0	13.19	99.62	13.19	99.62
-270.0	13.19	99.68	13.19	99.68
-255.0	13.20	99.72	13.20	99.72
-240.0	13.20	99.77	13.20	99.77
-225.0	13.20	99.80	13.20	99.80
-210.0	13.20	99.83	13.20	99.83
-195.0	13.20	99.85	13.20	99.85
-180.0	13.21	99.87	13.21	99.87
-165.0	13.21	99.89	13.21	99.89
-150.0	13.21	99.90	13.21	99.90
-135.0	13.21	99.91	13.21	99.91
-120.0	13.21	99.92	13.21	99.92
-105.0	13.21	99.93	13.21	99.93
-90.0	13.21	99.94	13.21	99.94
-75.0	13.21	99.95	13.21	99.95
-60.0	13.21	99.96	13.21	99.96
-45.0	13.21	99.96	13.21	99.96
-30.0	13.21	99.97	13.21	99.97
-15.0	13.21	99.97	13.21	99.97
Unsteady computation begin:				
0.0	13.21	99.97	13.21	99.97
15.0	13.21	99.98	13.21	99.98
30.0	13.21	99.98	13.21	99.98
45.0	13.22	100.15	13.22	100.15

60.0	13.25	101.76	13.25	101.76
75.0	13.39	108.55	13.39	108.55
90.0	13.77	125.56	13.77	125.56
105.0	14.45	152.56	14.45	152.56
120.0	15.04	182.99	15.04	167.67
135.0	15.36	195.16	15.36	168.83
150.0	15.45	193.59	15.45	163.93
165.0	15.39	184.03	15.39	156.02
180.0	15.25	170.92	15.25	147.31
195.0	15.08	159.13	15.08	140.29
210.0	14.91	149.04	14.91	134.52
225.0	14.75	140.71	14.75	129.86
240.0	14.60	133.90	14.60	126.06
255.0	14.47	128.27	14.47	122.86
270.0	14.36	123.74	14.36	120.25
285.0	14.25	119.96	14.25	117.96
300.0	14.15	116.93	14.15	116.03
315.0	14.06	114.41	14.06	114.28
330.0	13.98	112.41	13.98	112.67
345.0	13.90	110.92	13.90	110.92
360.0	13.82	109.51	13.82	109.51
375.0	13.76	108.29	13.76	108.29
390.0	13.69	107.23	13.69	107.23
405.0	13.64	106.31	13.64	106.31
420.0	13.59	105.50	13.59	105.50

ELEVATION AND DISCHARGE ALONG THE CHANNEL

i	X(m)	YMAX(m)	Minute	Y(m)	Q(cms)	Minute
1	0.00	34.55	75.00	24.05	100.00	420.00
2	1000.00	32.19	75.00	22.99	100.13	420.00
3	2000.00	30.05	90.00	21.98	100.31	420.00
4	3000.00	28.26	90.00	21.04	100.52	420.00
5	4000.00	26.52	105.00	20.14	100.77	420.00
6	5000.00	25.04	105.00	19.28	101.08	420.00
7	6000.00	23.58	120.00	18.46	101.44	420.00
8	7000.00	22.33	120.00	17.68	101.85	420.00
9	8000.00	21.08	120.00	16.93	102.32	420.00
10	9000.00	19.95	135.00	16.21	102.85	420.00
11	10000.00	18.84	135.00	15.51	103.44	420.00
12	11000.00	17.71	135.00	14.84	104.08	420.00
13	12000.00	16.59	150.00	14.20	104.77	420.00
14	13000.00	15.45	150.00	13.59	105.50	420.00
15	13000.00	15.45	150.00	13.59	105.50	420.00
16	14000.00	14.62	165.00	13.00	106.27	420.00
17	14000.00	14.32	165.00	12.64	106.27	420.00
18	15000.00	13.52	165.00	12.03	107.25	420.00
19	16000.00	12.72	180.00	11.41	108.29	420.00
20	17000.00	11.96	195.00	10.80	109.40	420.00
21	18000.00	11.23	210.00	10.21	110.57	420.00
22	19000.00	10.55	210.00	9.62	111.81	420.00
23	20000.00	9.91	225.00	9.06	113.10	420.00
24	21000.00	9.31	240.00	8.50	114.46	420.00
25	22000.00	8.73	240.00	7.96	115.87	420.00
26	23000.00	8.19	240.00	7.43	117.33	420.00
27	24000.00	7.68	240.00	6.93	118.84	420.00

## PROGRAM KOMPUTER ATP1DUST

Di bawah ini disajikan program komputer ATP1DUST untuk menyelesaikan hitungan aliran tak permanen untuk saluran tunggal, beserta dengan file yang berisi COMMON variabel yang digunakan dalam program dan struktur data masukan. Program ini dibuat general untuk menangani beberapa jenis kondisi batas hulu, hilir dan internal. Nama program adalah ATP1DUST yang merupakan kependekan dari aliran tak permanen 1-D untuk saluran tunggal. Pemrograman dilakukan dengan bahasa FORTRAN 77 standar, sehingga kompatibilitas tinggi untuk berbagai jenis FORTRAN 77 kompilator. Ada beberapa kompilator yang tidak mampu menerima perintah INCLUDE, jika terjadi yang demikian maka file COMMON harus langsung disisipkan di tempat dimana terdapat perintah INCLUDE. Program ini ditulis secara umum mengikuti bagan alir dari metoda 'sapan-ganda' untuk saluran/sungai tunggal seperti disajikan dalam Gambar 4.

### STRUKTUR DATA MASUKAN:

File yang berisi struktur data masukan mempunyai format khusus, sehingga tidak akan disajikan disini, tetapi akan disajikan sebagai lampiran.

### FILE 'commons':

```

C-----6-----1-----2-----3-----4-----5-----6-----77
COMMON/GLOBAL/THETA, ALPHA, BETA, GRAV, DT, TIMES, TIMEH, TIMEH
COMMON/CHANNEL/LENGTH, SLOPE, STRICKLER
COMMON/INOUTFILES/INF, OUTF

REAL      LENGTH
INTEGER   INF, OUTF
C-----6-----1-----2-----3-----4-----5-----6-----77

```

### PROGRAM KOMPUTER:

```

C0***6****1*****2*****3*****4*****5*****6*****77
C
C   Program untuk menghitung aliran tak permanen 1-D untuk saluran
C   tunggal dengan beberapa "internal boundary conditions" dan
C   beberapa kondisi batas sbb:
C   Internal boundary conditions:
C       1. Inflow/outflow from outside
C       2. Lateral Storage Pocket
C       3. Sluice gate
C       4. Weir
C       5. Freeflow intake
C   Kondisi batas: 1. Q(t), hidrograp debit
C                  2. Q(y), "rating curve"
C                  3. Q = K*sqrt(S0), "Locally uniform flow"
C                  4. Y(t), hidrograp elevasi muka air
C
C   "Linear Solver" nya menggunakan metoda "double sweep."
C

```

```

C                               diprogram oleh:                               C
C                               Djoko Luknanto                               C
C                               Akhir Nopember 1991                          C
C                               Version 1.01 - Nov.1991                       C
C                               C                                             C
C0***6***1*****2*****3*****4*****5*****6*****77

```

```
PROGRAM ATP1DUST
```

```
PARAMETER (MAXARRAY = 100, MAXIBC = 10, NUMOFIBC = 5)
```

```
INCLUDE 'commons'
```

```

REAL      Q(MAXARRAY), Y(MAXARRAY), E(MAXARRAY), F(MAXARRAY),
1         L(MAXARRAY), M(MAXARRAY), N(MAXARRAY), BCUS(MAXARRAY),
2         DBCUS(MAXARRAY), BCDS(MAXARRAY), DBCDS(MAXARRAY),
3         SURF(MAXARRAY), X(MAXARRAY), YMAX(MAXARRAY),
4         TYMAX(MAXARRAY), QINFLOW(MAXARRAY),
5         K, K1
INTEGER   IBCTYP(MAXIBC), IIBC(MAXIBC), IPRINT(MAXARRAY),
1         MAP(MAXARRAY), IWORK(MAXARRAY),
2         CONSDX
CHARACTER FILENAME(2)*20, TITLE(5)*60, TAB*1
LOGICAL  STEADY, FIRST

```

```
-----
UTILITIES CONSTANTS
-----
```

```
TAB      = CHAR(9)
```

```
-----
REQUEST INPUT & OUTPUT FILES
-----
```

```

INF      = 35
OUTF     = 36
CALL OPENFILES(INF,OUTF,FILENAME)

```

```
-----
READING GENERAL DATA: RECORD 1 AND 2
-----
```

```

READ(INF,*) NTITLE
IF (NTITLE.GT.5) STOP 'Title should be less than 5 lines !'
DO 5 I = 1, NTITLE
5     READ (INF,'(A60)') TITLE(I)
READ(INF,*) GRAV, ALPHA, BETA, THETA, TOTIMEM, DTIMEM,
1     STEADY, EPS, MAXCOUNT
MAXCOUNT = ABS(MAXCOUNT)
EPS = ABS(EPS)

```

```
-----
READING REACH DATA: RECORD 3
-----
```

```

READ(INF,*) CONSDX
IF (CONSDX.LE.0) THEN
    READ(INF,*) NDX, DX
    IF (NDX.GT.MAXARRAY) STOP 'NDX exceeded MAXARRAY !'
    DO 6 I = 1,NDX
6     X(I) = FLOAT(I-1)*DX
ELSE IF (CONSDX.EQ.1) THEN
    READ(INF,*) NDX, (X(I), I = 1,NDX)
    IF (NDX.GT.MAXARRAY) STOP 'NDX exceeded MAXARRAY !'
ELSE IF (CONSDX.GE.2) THEN
    READ(INF,*) NXREGION, XUS
    NDX = 1
    DO 8 I = 1, NXREGION
        READ(INF,*) XDS, NREACH
        DX = (XDS-XUS)/NREACH
        DO 7 J = 1, NREACH
            X(NDX) = XUS

```

```

        NDX      = NDX + 1
        IF (NDX.GT.MAXARRAY) STOP 'NDX exceeded MAXARRAY !'
        XUS      = XUS + DX
7         CONTINUE
8         CONTINUE
        X(NDX) = XUS
ENDIF

C -----
C READING CHANNEL DATA: RECORD 4
C -----
READ(INF,*) STRICKLER, SLOPE

C -----
C READING INITIAL DATA: RECORD 5
C -----
READ(INF,*) INITCONS
IF (INITCONS.EQ.1) THEN
    READ(INF,*) QINIT, YINIT
ELSE IF (INITCONS.EQ.2) THEN
    READ(INF,*) (Q(I),Y(I), I = 1,NDX)
ELSE IF (INITCONS.EQ.3) THEN
    READ(INF,*) QINIT, NYREGION, YINITUS
    IY = 1
    DO 11 I = 1, NYREGION
        READ(INF,*) YINITDS, NREACH
        DY = (YINITUS-YINITDS)/NREACH
        DO 9 J = 1, NREACH
            Y(IY) = YINITUS
            Q(IY) = QINIT
            IY = IY + 1
            IF (IY.GT.MAXARRAY) STOP 'IY exceeded MAXARRAY !'
        YINITUS = YINITUS - DY
9         CONTINUE
11        CONTINUE
        Y(IY) = YINITUS
        Q(IY) = QINIT
ENDIF

C -----
C READING PRINTING CONTROL: RECORD 6
C -----
READ(INF,*) IFREQ, NPRINT, (IPRINT(I), I = 1,NPRINT)

LENGTH = X(NDX)-X(1)
NDT     = TOTIMEM/DTIMEM

C -----
C READING U/S B.C.: RECORD 7
C -----
READ(INF,*) IUSTYP
IF (IUSTYP.LE.0 .OR. IUSTYP.GE.5) THEN
    WRITE(*,'(A21,I2)') ' Undefined U/S B.C. =',IUSTYP
    STOP 'Program terminated'
ELSE IF (IUSTYP.NE.3) THEN
    READ(INF,*) NBCUS, (BCUS(I), I=1,NBCUS)
    Compute the derivative
    IF (IUSTYP.EQ.2) CALL PRIME(BCUS,NBCUS,DBCUS)
ENDIF

C -----
C READING D/S B.C.: RECORD 8
C -----
READ(INF,*) IDSTYP
IF (IDSTYP.LE.0 .OR. IDSTYP.GE.5) THEN
    WRITE(*,'(A21,I2)') ' Undefined D/S B.C. =',IDSTYP
    STOP 'Program terminated'
ELSE IF (IDSTYP.NE.3) THEN
    READ(INF,*) NBCDS, (BCDS(I), I=1,NBCDS)
    Compute the derivative

```

```

      IF (IDSTYP.EQ.2) CALL PRIME(BCDS,NBCDS,DBCDS)
ENDIF

C -----
C READING INTERNAL B.C.: RECORD 9 AND 10
C -----
READ(INF,*) NIBC
IF (NIBC.GT.MAXIBC) STOP 'Number of IBCs exceeded MAXIBC !'
IF ( (NIBC+NDX).GT.MAXARRAY )
1  STOP 'NDX + NIBC exceeded MAXARRAY !'

DO 100 I = 1, NIBC

  READ(INF,*) LIBC, IBC
  IBCTYP(I) = IBC
  IIBC(I) = LIBC

  IF (IBC.GT.NUMOFIBC .OR. IBC.LE.0) THEN
    WRITE(*,'(A26,I2)') ' Undefined internal b.c. =', IBC
    WRITE(*,'(A26,I2)') ' At internal b.c. number =', I
    STOP 'Program terminated'
  ENDIF

  GOTO (10,20,30,40,50) IBC

10  READ(INF,*) NINFLOW, (QINFLOW(J), J = 1, NINFLOW)
    IF (NINFLOW.GT.MAXARRAY) STOP 'NINFLOW exceeded MAXARRAY !'
    GOTO 100

20  READ(INF,*) FLOORELEV, NSURF, (SURF(J), J = 1, NSURF)
    IF (NSURF.GT.MAXARRAY) STOP 'NSURF exceeded MAXARRAY !'
    GOTO 100

30  READ(INF,*) CGATE, AGATE
    GOTO 100

40  READ(INF,*) CWEIR, BWEIR, YWEIR, HWEIR
    GOTO 100

50  READ(INF,*) CINTAKE, BINTAKE, YINTAKE

100 CONTINUE

C -----
C CLOSE THE INPUT FILE
C -----
CLOSE (INF)

C -----
C INITIALIZATION
C -----
CALL RAZCONS (L,NDX,0.0)
CALL RAZCONS (M,NDX,0.0)
CALL RAZCONS (N,NDX,0.0)
CALL RAZCONS (E,NDX,0.0)
CALL RAZCONS (F,NDX,0.0)
STEADY = .NOT.STEADY
ICOUNT = 0
FIRST = .TRUE.

C -----
C INITIAL CONDITION
C -----
IF (INITCONS.EQ.1) THEN
  DO 105 I = 1,NDX
    Q(I) = QINIT
    Y(I) = YINIT
105  CONTINUE
ELSE IF (INITCONS.GE.4) THEN
  CALL INCOND (NDX, X, Y, Q)

```



```

ENDIF

C -----
C MANAGEMENT OF INDICES OF INTERNAL B.C.
C -----
CALL SHIFTER (NDX, NIBC, X, Y, Q, IIBC, MAP)

C -----
C MANAGEMENT OF OUTPUT CONTROL
C -----
CALL PRINTSHIFTER (NDX, NPRINT, IPRINT, MAP, IWORK, MAXARRAY)

C -----
C PRINT HEADER & INFORMATIONS
C -----
CALL HEADER (NTITLE, TITLE, FILENAME, TOTIMEM, DTIMEM, NDT,
1     NDX, STEADY, EPS, MAXCOUNT, INITCONS, QINIT, YINIT,
2     IFREQ, NPRINT, IPRINT, IUSTYP, IDSTYP, NIBC, IBCTYP,
3     IIBC, FLOORELEV, CGATE, AGATE, CWEIR, BWEIR, YWEIR, HWEIR,
4     CINTAKE, BINTAKE, YINTAKE, X)

C -----
C INITIALIZE Y MAXIMUM
C -----
DO 120 I = 1, NDX
    YMAX(I) = 0.0
    TYMAX(I) = 0.0
120 CONTINUE

C -----
C DT MUST BE IN SECOND IN ORDER TO
C CONFORM WITH DISCHARGE (M3/SEC)
C -----
DT = DTIMEM*60.0
TIMES = -DT

C -----
C Print the output for the first time
C -----
IF (.NOT. STEADY)WRITE (*, '(A)') 'Steady state stabilization:'

C -----
C LOOP WITH TIME
C -----
DO 1000 IT = 0, NDT

    TIMES = TIMES + DT
    TIMEM = TIMES/60.0
    TIMEH = TIMEM/60.0

C     Compute geometry of the first cross-section
C     -----
500 CALL GEOMETRY(X(1),Y(1), AREA, WIDTH, K, DKDY)

    IF (IUSTYP.EQ.4) THEN

        USGAMMA = TABINT(TIMEM,BCUS,NBCUS) - Y(1)
        C     Compute geometry of the second cross-section
        C     -----
        CALL GEOMETRY (X(2),Y(2), AREAl, WIDTH1, K1, DKDY1)

C     Compute influence coefficient
C     -----
        DX = X(2) - X(1)
        CALL UNSTEADYCOEFF (DX, WIDTH, WIDTH1, Y(1), Y(2),
1     AREA, AREAl, Q(1), Q(2), K, K1, DKDY, DKDY1,
2     A, B, C, D, G, AA, BB, CC, DD, GG)
        DENOM = D*CC-C*DD
        L(1) = (A*CC-C*AA)/DENOM
        M(1) = (B*CC-C*BB)/DENOM

```

```

N(1) = (C*GG-G*CC)/DENOM
C
C Initialize E(2) & F(2) from d/s b.c.
C -----
DENOM = D*BB-B*DD
E(2) = (A*DD-D*AA)/DENOM
F(2) = (D*(CC*USGAMMA+GG)-DD*(C*USGAMMA+G))/DENOM
C
C Store geometry elements for the next section
C -----
WIDTH = WIDTH1
AREA = AREA1
K = K1
DKDY = DKDY1
C
C Initialize the start index of forward sweep
C -----
ISTART = 2
ELSE
C Initialize E(1) & F(1) from d/s b.c.
C -----
1 CALL USBC (IUSTYP, X(1), Y(1), Q(1), NBCUS, BCUS, DBCUS,
E(1), F(1))
C
C Initialize the start index of forward sweep
C -----
ISTART = 1
ENDIF
C
C FORWARD SWEEP
C =====
IBCDONE = 0
DO 200 I = ISTART, NDX-1
C
C Compute geometry of the cross-section
C -----
CALL GEOMETRY (X(I+1), Y(I+1), AREA1, WIDTH1, K1, DKDY1)
C
C Check for internal boundary conditions
C -----
IF (IBCDONE.GE.NIBC) GOTO 185
DO 180 J = 1,NIBC
C Check the location of the structure
C -----
IF (I.NE.IIBC(J)) GOTO 180
1 CALL IBCMANAGER (IBCTYP(J), Y(I), Y(I+1), Q(I), Q(I+1),
2 TIMEM, QINFLOW, NINFLOW, SURF, NSURF,
3 FLOORELEV, CGATE, AGATE, CWEIR, BWEIR, YWEIR,
4 CINTAKE, BINTAKE, YINTAKE,
A, B, C, D, G, AA, BB, CC, DD, GG)
IBCDONE = IBCDONE + 1
GOTO 190
180 CONTINUE
C
C Compute influence coefficient
C -----
185 DX = X(I+1) - X(I)
CALL UNSTEADYCOEFF (DX, WIDTH, WIDTH1, Y(I), Y(I+1),
1 AREA, AREA1, Q(I), Q(I+1), K, K1, DKDY, DKDY1,
2 A, B, C, D, G, AA, BB, CC, DD, GG)
190 CALL INFLUENCECOEFF (A, B, C, D, G, AA, BB, CC, DD, GG,
1 E(I), F(I), E(I+1), F(I+1), L(I), M(I), N(I))
C
C Store geometry elements for the next section
C -----
WIDTH = WIDTH1
AREA = AREA1
K = K1
DKDY = DKDY1

```

```

200      CONTINUE

C      Initialize DY & DQ from u/s b.c.
C      -----
1      CALL DSBC (IDSTYP, X(NDX), Y(NDX), Q(NDX), NBCDS, BCDS, DBCDS,
           E(NDX), F(NDX), DY, DQ)

C      RETURN SWEEP
C      =====
SUMDQ = 0.0
DO 300 I = NDX-1,1,-1
  Y(I+1) = Y(I+1) + DY
  Q(I+1) = Q(I+1) + DQ
  DY     = L(I)*DY + M(I)*DQ + N(I)
  DQ     = E(I)*DY + F(I)
  SUMDQ = SUMDQ + DQ*DQ
300    CONTINUE
      IF (IUSTYP.EQ.4) THEN
        Y(1) = Y(1) + USGAMMA
        Q(1) = Q(1) + DY
        SUMDQ = SUMDQ + DY*DY
      ELSE
        Y(1) = Y(1) + DY
        Q(1) = Q(1) + DQ
        SUMDQ = SUMDQ + DQ*DQ
      ENDIF

C      Dump to screen
C      -----
1      IF (.NOT.FIRST)
2      WRITE (*, '(F8.1,20(A,F10.2))')
        TIMEM,(' ',Y(J),' ',Q(J),J=1,NDX,NDX-1)

C      Check for steady intial condition
C      -----
      IF (STEADY) GOTO 900
      IF (SQRT(SUMDQ/NDX).GT.EPS .AND. ICOUNT.LE.MAXCOUNT) THEN
        ICOUNT = ICOUNT + 1
        FIRST = .FALSE.
        GOTO 500
      ELSE
        STEADY= .TRUE.
        FIRST = .TRUE.
      ENDIF

C      Print the output
C      -----
900    IF (FIRST) THEN
1      WRITE (OUTF, '(A,I6,A)')
        'Steady state stabilization:', ICOUNT, ' time steps'
        WRITE (OUTF,1500) ('-----', I=1,NPRINT)
        WRITE (OUTF, '(A)') 'Unsteady computation begin:'

1      WRITE (*, '(A,I6,A)')
        'Steady state stabilization:', ICOUNT, ' time steps'
        WRITE (*, '(A)') 'Unsteady computation begin:'
        WRITE (*, '(F8.1,20(A,F10.2))')
1      TIMEM,(' ',Y(J),' ',Q(J),J=1,NDX,NDX-1)
        FIRST = .FALSE.
      ENDIF
      IF (MOD(IT,IFREQ).EQ.0) WRITE (OUTF, '(F6.1,20(A,F10.2))')
1      TIMEM,(TAB,Y(IPRINT(J)), TAB,Q(IPRINT(J)),J=1,NPRINT)

C      Compute Y MAXIMUM when unsteady computation start
C      -----
DO 350 I = 1,NDX
  IF (YMAX(I).LT.Y(I)) THEN
    YMAX(I) = Y(I)
  
```

```

          TYMAX(I) = TIMEM
        ENDIF
350      CONTINUE

1000    CONTINUE

C       Print Y maximum
C       -----
        WRITE(OUTF,1500) ('-----', I=1,NPRINT)
        WRITE(*,1501) ('-----', I=1,NDX,NDX-1)
1500    FORMAT ('-----',10A)
1501    FORMAT ('-----',10A)
        WRITE (OUTF,1600)
1600    FORMAT (//,18X,'ELEVATION AND DISCHARGE ALONG THE CHANNEL',/,
1         40(1H-),'|',35(1H-),/, ' i           X(m)           YMAX(m)',
2         '      Minute           Y(m)           Q(cms)           Minute',
3         /,40(1H-),'|',35(1H-))
        DO 2000 I = 1, NDX
2000    WRITE(OUTF,'(I4,6(A,F8.2))') I, TAB, X(I), TAB, YMAX(I),
1         TAB, TYMAX(I), TAB, Y(I), TAB, Q(I), TAB,
2         TIMEM
        WRITE(OUTF,1700)
1700    FORMAT (40(1H-),'|',35(1H-))

C       -----
C       CLOSE THE OUTPUT FILE
C       -----
        CLOSE (OUTF)

        STOP 'Job well done sir !'
        END

C0***6****1*****2*****3*****4*****5*****6*****77
        SUBROUTINE INCOND (NDX, X, Y, Q)
C0---6---1-----2-----3-----4-----5-----6-----77
        DIMENSION X(NDX), Y(NDX), Q(NDX)

        DO 110 I = 1,NDX
            XX = X(I)/1000.0
            Q(I) = 100.0
            Y(I) = 23.741 - 0.9653*XX + 0.0097*XX**2
110    CONTINUE

        RETURN
        END

C0***6****1*****2*****3*****4*****5*****6*****77
        SUBROUTINE UNSTEADYCOEFF (DX, WIDTH, WIDTH1, Y, Y1, AREA, AREA1,
1         Q, Q1, K, K1,DKDY, DKDY1, A, B, C, D, G, AA, BB, CC,
2         DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C       Subroutine untuk menghitung koefisien pengaruh dari
C       metoda "double sweep," A, B, C, D, G, AA, BB, CC, DD, GG
C       for full unsteady dynamic equation
C
C0---6---1-----2-----3-----4-----5-----6-----77

        INCLUDE 'commons'

        REAL K, K1

        A = 0.5*WIDTH1/DT
        B = THETA/DX
        C = -0.5*WIDTH/DT
        D = B
        G = (Q-Q1)/DX

        BETA1 = 1.0 - BETA
        V = Q/AREA

```

```

V1    = Q1/AREA1
QK    = Q*ABS(Q)/K**2
QK1   = Q1*ABS(Q1)/K1**2
ASUM  = AREA+AREA1

```

```

TERM2 = ALPHA*THETA*WIDTH1*V1*(Q-Q1)/(DX*AREA1)
TERM3 = -0.25*ALPHA*THETA*WIDTH1*(V+V1)*
1      (V+V1*(2.0*AREA/AREA1-1.0))/DX
TERM4 = 0.5*THETA*GRAV*(ASUM+WIDTH1*(Y1-Y))/DX
TERM5 = 0.5*THETA*GRAV*WIDTH1*(BETA1*QK1+BETA*QK) -
1      BETA1*THETA*GRAV*DKDY1*ASUM*QK1/K1
AA    = TERM2 + TERM3 + TERM4 + TERM5

```

```

TERM1 = 0.5/DT
TERM2 = ALPHA*THETA*(V-Q/AREA1+2.0*V1)/DX
TERM3 = 0.5*ALPHA*THETA*(V+V1)*(AREA/AREA1-1.0)/DX
TERM5 = BETA1*THETA*GRAV*ASUM*QK1/Q1
BB    = TERM1 + TERM2 + TERM3 + TERM5

```

```

TERM2 = ALPHA*THETA*WIDTH*V*(Q1-Q)/(DX*AREA)
TERM3 = 0.25*ALPHA*THETA*WIDTH*(V+V1)*
1      (V*(1.0-2.0*AREA1/AREA)-V1)/DX
TERM4 = 0.5*THETA*GRAV*(ASUM+WIDTH*(Y-Y1))/DX
TERM5 = -0.5*THETA*GRAV*WIDTH*(BETA1*QK1+BETA*QK) +
1      BETA*THETA*GRAV*DKDY*ASUM*QK/K
CC    = TERM2 + TERM3 + TERM4 + TERM5

```

```

TERM1 = -0.5/DT
TERM2 = ALPHA*THETA*(2.0*V-Q1/AREA+V1)/DX
TERM3 = 0.5*ALPHA*THETA*(V+V1)*(AREA1/AREA-1.0)/DX
TERM5 = -BETA*THETA*GRAV*ASUM*QK/Q
DD    = TERM1 + TERM2 + TERM3 + TERM5

```

```

TERM2 = ALPHA*(Q-Q1)*(V+V1)/DX
TERM3 = 0.25*ALPHA*(AREA1-AREA)*(V+V1)**2/DX
TERM4 = 0.5*GRAV*ASUM*(Y-Y1)/DX
TERM5 = -0.5*GRAV*ASUM*(BETA1*QK1+BETA*QK)
GG    = TERM2 + TERM3 + TERM4 + TERM5

```

```

RETURN
END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77
  SUBROUTINE INFLUENCECOEFF (A, B, C, D, G, AA, BB, CC, DD, GG,
1      E, F, E1, F1, L, M, N)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C      Subroutine untuk menghitung koefisien pengaruh dari
C      metoda "double sweep," E, F, L, M, N.
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

REAL L, M, N

```

```

DENOM = C*DD-CC*D
L      = (A*DD-AA*D)/DENOM
M      = (B*DD-BB*D)/DENOM
N      = (D*GG-DD*G)/DENOM

CDE    = C+D*E
DENOM  = B-M*CDE
E1     = (L*CDE-A)/DENOM
F1     = (N*CDE+D*F+G)/DENOM

```

```

RETURN
END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77

```

```

SUBROUTINE GEOMETRYDWOPER(X, Y, A, B, K, DKDY)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Subroutine untuk menghitung LUAS, LEBAR MUKA AIR, CONVEYANCE dan
C   DK/DY di titik I.
C
C0---6---1-----2-----3-----4-----5-----6-----77

  INCLUDE 'commons'

  REAL K, M

  M      = 5.0
  WIDBOT = 121.92
  YBOT   = SLOPE*(LENGTH-X)
  HEIGHT = Y-YBOT

  B      = WIDBOT + 2.*M*HEIGHT
  P      = WIDBOT + 2.*SQRT(1.+M*M)*HEIGHT
  PPRIME = 2.*SQRT(1.+M*M)
  A      = (B+WIDBOT)*HEIGHT*0.5
  APRIME = (B+WIDBOT)*0.5
  K      = STRICKLER*A**(5./3.)/P**(2./3.)
  DKDY   = STRICKLER*( 5./3.*A**(2./3.)*APRIME/P**(2./3.) -
1         2./3./P**(5./3.)*PPRIME*A**(5./3.) )

  RETURN
  END

```

```

C0***6***1*****2*****3*****4*****5*****6*****77
SUBROUTINE GEOMETRY (X, Y, A, B, K, DKDY)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Subroutine untuk menghitung LUAS, LEBAR MUKA AIR, CONVEYANCE dan
C   DK/DY di titik I.
C
C0---6---1-----2-----3-----4-----5-----6-----77

  INCLUDE 'commons'

  REAL K

  WIDTHUS = 8.0
  WIDTHDS = 20.0

  WIDBOT = WIDTHUS + X/LENGTH*(WIDTHDS-WIDTHUS)
  YBOT   = SLOPE*(LENGTH-X)
  B      = WIDBOT
  P      = WIDBOT + 2.0*(Y-YBOT)
  PPRIME = 2
  A      = B*(Y-YBOT)
  APRIME = B
  K      = STRICKLER*A**(5./3.)/P**(2./3.)
  DKDY   = STRICKLER*( 5./3.*A**(2./3.)*APRIME/P**(2./3.) -
1         2./3./P**(5./3.)*PPRIME*A**(5./3.) )

  RETURN
  END

```

```

C0*****1*****2*****3*****4*****5*****6*****77
FUNCTION TABINT(T,TAB,LENGTH)
C-----
C
C   LINEAR INTERPOLATION OF DATA-PAIRS IN THE TAB(LENGTH)
C
C   ON INPUT :
C   TAB contains data as follows :
C   T1,F1,T2,F2,           ,Tn,Fn
C
C   ON RETURN :

```

```

C          TABINT return the value of F at time T

C          NOTE :
C          If T <= T1 then TABINT = F1
C          If T >= Tn then TABINT = Fn

C-----
          DIMENSION TAB(LENGTH)

          IF (T.LE.TAB(1)) THEN
            TABINT = TAB(2)
            GO TO 999
          ELSE
            IF (T.GT.TAB(LENGTH-1)) GO TO 100
          ENDIF

          DO 50 I=1,LENGTH,2
            IF (TAB(I).GT.T) THEN
              J=I-2
              TABINT=TAB(J+1)+(TAB(J+3)-TAB(J+1))*(T-TAB(J))/(TAB(J+2)-TAB(J))
              GO TO 999
            ENDIF
          50 CONTINUE

          100 TABINT = TAB(LENGTH)

          999 RETURN
          END

C0*****1*****2*****3*****4*****5*****6*****77
          SUBROUTINE PRIME(TAB,LENGTH,TABPRIME)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C          Menghitung turunan pertama dari tabel ke sumbu mendatar
C
C0---6---1-----2-----3-----4-----5-----6-----77
          DIMENSION TAB(LENGTH), TABPRIME(LENGTH-2)
          REAL NUMERATOR

          DO 10 I=1,LENGTH-2,2

            TABPRIME(I) = TAB(I)
            NUMERATOR = TAB(I+3)-TAB(I+1)
            DENOMINATOR = TAB(I+2)-TAB(I)
            IF (ABS(DENOMINATOR).LE.1.0E-4.AND.ABS(NUMERATOR).LE.0.01) THEN
              TABPRIME(I+1) = 0.0
            ELSE
              TABPRIME(I+1) = NUMERATOR/DENOMINATOR
            ENDIF

          10 CONTINUE

          RETURN
          END

C0***6***1*****2*****3*****4*****5*****6*****77
          SUBROUTINE USBC (NBC, X1, Y1, Q1, NDATA, TAB, TABPRIME, E1, F1)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C          "Initialize the coefficients of double sweep for the first time"
C          untuk beberapa jenis kondisi batas hulu sbb:
C
C          NBC: 1 KONDISI BATAS HULU: Q(t), hidrograp debit
C                2 Q(y), "rating curve"
C                3 Q = K*sqrt(S0), "Locally uniform flow"
C                4 Kondisi batas hulu y(t) dihitung dalam program utama
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

INCLUDE 'commons'

DIMENSION TAB(1), TABPRIME(1)
REAL K

IF (NBC.LE.0 .OR. NBC.GE.4) RETURN
GOTO (100,200,300) NBC

100  E1 = 0.0
     F1 = TABINT(TIMEM,TAB,NDATA) - Q1
     RETURN

200  E1 = TABINT(Y1,TABPRIME,NDATA-2)
     F1 = TABINT(Y1,TAB,NDATA) - Q1
     RETURN

300  CALL GEOMETRY(X1, Y1, A, B, K, DKDY)
     E1 = DKDY*SQRT(SLOPE)
     F1 = K*SQRT(SLOPE) - Q1

     RETURN
     END

C0***6****1*****2*****3*****4*****5*****6*****77
      SUBROUTINE DSBC (NBC, XN, YN, QN, NDATA, TAB, TABPRIME, EN, FN,
1         1             DY, DQ)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   "Initialize the coefficients of double sweep for the first time"
C   untuk beberapa jenis kondisi batas hilir sbb:
C
C   NBC: 1  KONDISI BATAS HULU: Q(t), hidrograp debit
C         2                    Q(y), "rating curve"
C         3                    Q = K*sqrt(S0), "Locally uniform flow"
C         4                    y(t), hidrograp elevasi muka air
C
C0---6---1-----2-----3-----4-----5-----6-----77

INCLUDE 'commons'

DIMENSION TAB(1), TABPRIME(1)
REAL K

IF (NBC.LE.0 .OR. NBC.GE.5) RETURN
GOTO (100,200,300,400) NBC

100  DQ = TABINT(TIMEM,TAB,NDATA) - QN
     DY = (DQ-FN)/EN
     RETURN

200  ALFA = TABINT(YN,TABPRIME,NDATA-2)
     GAMMA = QN - TABINT(YN,TAB,NDATA)
     DY = (GAMMA + FN)/(ALFA - EN)
     DQ = EN*DY + FN
     RETURN

300  CALL GEOMETRY(XN, YN, A, B, K, DKDY)
     ALFA = DKDY*SQRT(SLOPE)
     GAMMA = QN - K*SQRT(SLOPE)
     DY = (GAMMA + FN)/(ALFA - EN)
     DQ = EN*DY + FN
     RETURN

400  DY = TABINT(TIMEM,TAB,NDATA) - YN
     DQ = EN*DY + FN

     RETURN
     END

C0***6****1*****2*****3*****4*****5*****6*****77

```



```

SUBROUTINE INOUTFLOW (Y, Y1, Q, Q1, TIMEM, QINFLOW, NINFLOW,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Untuk menghitung koefisien A, B, C, D, G, AA, BB, CC, DD, GG
C   karena Inflow/Outflow from outside
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

DIMENSION QINFLOW(NINFLOW)

```

```

A = 0.0
B = 1.0
C = 0.0
D = 1.0
G = Q - Q1 + TABINT(TIMEM, QINFLOW, NINFLOW)
AA = 1.0
BB = 0.0
CC = 1.0
DD = 0.0
GG = Y - Y1

```

```

RETURN
END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77
SUBROUTINE STORAGE (Y, Y1, Q, Q1, SURF, NSURF, FLOORELEV,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Untuk menghitung koefisien A, B, C, D, G, AA, BB, CC, DD, GG
C   karena Storage pocket
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

INCLUDE 'commons'

```

```

DIMENSION SURF(1)

```

```

IF (Y.GT.FLOORELEV) THEN
  A = 0.0
  B = -THETA
  C = TABINT(Y, SURF, NSURF)/DT
  D = -THETA
  G = Q1 - Q
ELSE
  A = 0.0
  B = 1.0
  C = 0.0
  D = 1.0
  G = Q1 - Q
ENDIF
AA = 1.0
BB = 0.0
CC = 1.0
DD = 0.0
GG = Y - Y1

RETURN
END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77
SUBROUTINE GATE (Y, Y1, Q, Q1, CGATE, AGATE,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Untuk menghitung koefisien A, B, C, D, G, AA, BB, CC, DD, GG
C   karena sluice gate
C

```

C0---6---1-----2-----3-----4-----5-----6-----77

INCLUDE 'commons'

PARAMETER (EPS = 0.01)

A = 0.0  
 B = 1.0  
 C = 0.0  
 D = 1.0  
 G = Q - Q1

CG = CGATE\*AGATE\*SQRT(2.0\*GRAV)

IF (ABS(Y-Y1).GE.EPS) THEN

C Use original formula  
 AA = -CG/(2.0\*SQRT(Y-Y1))  
 BB = 0.0  
 CC = AA  
 DD = 1.0  
 GG = Q - CG\*SQRT(Y-Y1)

ELSE

C Use approximation formula  
 AA = -CG/SQRT(EPS)  
 BB = 0.0  
 CC = AA  
 DD = 1.0  
 GG = Q - CG\*(Y-Y1)/SQRT(EPS)

ENDIF

RETURN

END

C0\*\*\*6\*\*\*1\*\*\*\*\*2\*\*\*\*\*3\*\*\*\*\*4\*\*\*\*\*5\*\*\*\*\*6\*\*\*\*\*77

SUBROUTINE WEIR (Y, Y1, Q, Q1, CWEIR, BWEIR, YWEIR,  
 1 A, B, C, D, G, AA, BB, CC, DD, GG)

C0---6---1-----2-----3-----4-----5-----6-----77

C  
 C Untuk menghitung koefisien A, B, C, D, G, AA, BB, CC, DD, GG  
 C karena weir

C  
 C0---6---1-----2-----3-----4-----5-----6-----77

INCLUDE 'commons'

PARAMETER (EPS = 0.01)

LOGICAL FREEFLOW

A = 0.0  
 B = 1.0  
 C = 0.0  
 D = 1.0  
 G = Q - Q1

IF (Y.GT.Y1) THEN

1 CALL WEIRCALC (CWEIR, BWEIR, YWEIR, Y, Y1, EPS, FREEFLOW,  
 QWEIR, QPRIMEUS, QPRIMEDS)

AA = QPRIMEDS  
 BB = 0.0  
 CC = -QPRIMEUS  
 DD = 1.0  
 GG = Q - QWEIR

ELSE

1 CALL WEIRCALC (CWEIR, BWEIR, YWEIR, Y1, Y, EPS, FREEFLOW,  
 QWEIR, QPRIMEUS, QPRIMEDS)

AA = -QPRIMEUS  
 BB = 0.0  
 IF (FREEFLOW) THEN  
 CC = -0.0001\*QPRIMEUS  
 ELSE

```

      CC = QPRIMEDS
      ENDIF
      DD = 1.0
      GG = Q + QWEIR
    ENDIF

```

```

  RETURN
  END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77
  SUBROUTINE WEIRCALC (M, B, YW, YUS, YDS, EPS, FREEFLOW,
  1                      QWEIR, QPRIMEUS, QPRIMEDS)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Untuk menghitung koefisien QWEIR, QPRIMEUS, QPRIMEDS
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

  INCLUDE 'commons'

```

```

  PARAMETER (SMALL = 0.001)

```

```

  LOGICAL FREEFLOW, NOFLOW
  REAL M

```

```

  FREEFLOW = (YDS - YW) .LE. (2.0/3.0*(YUS - YW))
  NOFLOW   = YW.GE.YUS .AND. YW.GE.YDS

```

```

C   IF (NOFLOW) THEN
C     No flow condition
C     QWEIR   = 0.0
C     QPRIMEUS = M*B*SQRT(2./3.*GRAV*SMALL)
C     QPRIMEDS = -SMALL*QPRIMEUS
C   ELSE IF (FREEFLOW) THEN
C     Free flowing weir
C     QWEIR   = 2./3.*M*B*SQRT(2./3.*GRAV)*(YUS-YW)**1.5
C     QPRIMEUS = M*B*SQRT(2./3.*GRAV*(YUS-YW))
C     QPRIMEDS = 0.0
C   ELSE IF (ABS(YUS-YDS).GE.EPS) THEN
C     Flooded weir original formula
C     QWEIR   = M*B*SQRT(2.*GRAV*(YUS-YDS))*(YDS-YW)
C     QPRIMEUS = M*B*SQRT(GRAV/2.0/(YUS-YDS))*(YDS-YW)
C     QPRIMEDS = M*B*SQRT(GRAV/2.0/(YUS-YDS))*(2.*YUS+YW-3.*YDS)
C   ELSE
C     Flooded weir approximation formula
C     QWEIR   = M*B*(YUS-YDS)*(YDS-YW)*SQRT(2.*GRAV/EPS)
C     QPRIMEUS = M*B*(YDS-YW)*SQRT(2.*GRAV/EPS)
C     QPRIMEDS = M*B*(YUS+YW-2.*YDS)*SQRT(2.*GRAV/EPS)
  ENDIF

  RETURN
  END

```

```

C0***6****1*****2*****3*****4*****5*****6*****77
  SUBROUTINE INTAKE (YUS, YDS, QUS, QDS, CINTAKE, BINTAKE, YINTAKE,
  1                      A, B, C, D, G, AA, BB, CC, DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C   Untuk menghitung koefisien A, B, C, D, G, AA, BB, CC, DD, GG
C   karena intake
C
C0---6---1-----2-----3-----4-----5-----6-----77

```

```

  INCLUDE 'commons'

```

```

  LOGICAL NOFLOW

```

```

  NOFLOW = YINTAKE.GE.YUS .AND. YINTAKE.GE.YDS

```

```

      IF (NOFLOW) THEN
C      No flow condition
        A = 0.0
        B = 1.0
        C = 0.0
        D = 1.0
        G = QUS - QDS
        AA = 1.0
        BB = 0.0
        CC = 1.0
        DD = 0.0
        GG = YUS - YDS
      ELSE
C      Free flowing weir
        QINTAKE = -2./3.*CINTAKE*BINTAKE*SQRT(2./3.*GRAV)*
1        (YUS-YINTAKE)**1.5
        QPRIME = -CINTAKE*BINTAKE*SQRT(2./3.*GRAV*(YUS-YINTAKE))

        A = 0.0
        B = 1.0
        C = QPRIME
        D = 1.0
        G = QUS - QDS + QINTAKE
        AA = 1.0
        BB = 0.0
        CC = 1.0
        DD = 0.0
        GG = YUS - YDS
      ENDIF

      RETURN
      END

C0***6****1*****2*****3*****4*****5*****6*****77
      SUBROUTINE IBCMANAGER (ID, Y, Y1, Q, Q1, TIMEM, QINFLOW, NINFLOW,
1      SURF, NSURF, FLOORELEV, CGATE, AGATE, CWEIR, BWEIR,
2      YWEIR, CINTAKE, BINTAKE, YINTAKE,
3      A, B, C, D, G, AA, BB, CC, DD, GG)
C0---6---1-----2-----3-----4-----5-----6-----77

      DIMENSION QINFLOW(1), SURF(1)

      GOTO (130,140,150,160, 170) ID

C      Inflow from outside
C      -----
130  CALL INOUTFLOW (Y, Y1, Q, Q1, TIMEM, QINFLOW, NINFLOW,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
      RETURN

C      Storage pocket
C      -----
140  CALL STORAGE (Y, Y1, Q, Q1, SURF, NSURF, FLOORELEV,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
      RETURN

C      Sluice gate
C      -----
150  CALL GATE (Y, Y1, Q, Q1, CGATE, AGATE,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
      RETURN

C      Weir
C      ----
160  CALL WEIR (Y, Y1, Q, Q1, CWEIR, BWEIR, YWEIR,
1      A, B, C, D, G, AA, BB, CC, DD, GG)
      RETURN

C      Freeflow intake

```

```

C -----
170 CALL INTAKE (Y, Y1, Q, Q1, CINTAKE, BINTAKE, YINTAKE,
1      A, B, C, D, G, AA, BB, CC, DD, GG)

      RETURN
      END

C0***6****1*****2*****3*****4*****5*****6*****77
      SUBROUTINE SHIFTER (NDX, NIBC, X, Y, Q, IIBC, IMAP)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C      To manage the shifting of indices (IIBC), length (X) due to
C      additional computational points for internal boundary condition
C
C0---6---1-----2-----3-----4-----5-----6-----77
      DIMENSION X(NDX+NIBC), Y(NDX+NIBC), Q(NDX+NIBC), IIBC(NIBC),
1      IMAP(NDX+NIBC)

      KSHIFT = NIBC
      KEND = NDX

      DO 1000 I = NDX,1,-1
        DO 100 J = 1,NIBC

C          Check the original place of IBC
          IF (IIBC(J).NE.I) GOTO 100

C          Start shifting from D/S to the place of IBC
          DO 200 K = KEND,I,-1
            ISHIFT = K + KSHIFT
            X(ISHIFT) = X(K)
            Y(ISHIFT) = Y(K)
            Q(ISHIFT) = Q(K)
            IMAP(ISHIFT) = K

200          CONTINUE
          KEND = I
          KSHIFT = KSHIFT - 1
          IIBC(J) = I + KSHIFT
          GOTO 1000

100          CONTINUE
1000         CONTINUE

      DO 2000 K = KEND,1,-1
2000        IMAP(K) = K

      NDX = NDX + NIBC

      RETURN
      END

C0***6****1*****2*****3*****4*****5*****6*****77
      SUBROUTINE PRINTSHIFTER(NDX, NPRINT, IPRINT, MAP, ITEMP, MAXARRAY)
C0---6---1-----2-----3-----4-----5-----6-----77
C
C      To manage the print indices after being shifted due to internal
C      boundary conditions.
C
C0---6---1-----2-----3-----4-----5-----6-----77

      DIMENSION IPRINT(1), MAP(1), ITEMP(1)

      ICOUNT = 0
      DO 45 J = 1, NPRINT
        DO 45 I = 1, NDX
          IF (MAP(I).EQ.IPRINT(J)) THEN
            ICOUNT = ICOUNT + 1
            IF (ICOUNT.GT.MAXARRAY) THEN
              ICOUNT = MAXARRAY
            
```

```

        WRITE (*,*)
        WRITE (*,*) ' Too many point to print !'
        WRITE (*,*) ' Only',ICOUNT,' points allowed'
        WRITE (*,*)
        GOTO 46
    ENDIF
    ITEMP(ICOUNT) = I
ENDIF
45     CONTINUE

46     NPRINT = ICOUNT
    DO 47 I = 1, NPRINT
47     IPRINT(I) = ITEMP(I)

    RETURN
    END

C0***6****1*****2*****3*****4*****5*****6*****77
SUBROUTINE HEADER (NTITLE, TITLE, FILENAME, TOTALMINUTES, DTMIN,
1     NDT, NDX, STEADY, EPS, MAXCOUNT, INITCONS,
2     QINIT, YINIT, IFREQ, NPRINT, IPRINT, IUSTYP, IDSTYP,
3     NIBC, IBCTYP, IIBC, FLOORELEV, CGATE, AGATE, CWEIR,
4     BWEIR, YWEIR, HWEIR, CINTAKE, BINTAKE, YINTAKE, X)
C0---6---1-----2-----3-----4-----5-----6-----77

    INCLUDE 'commons'

    DIMENSION IPRINT(NPRINT), IBCTYP(NIBC), IIBC(NIBC), X(NDX)
    CHARACTER TITLE(NTITLE)*60, BCNOTE(4)*50, FILENAME(2)*20
    LOGICAL STEADY

    BCNOTE(1) = 'Q(t), discharge given as a function of time      '
    BCNOTE(2) = 'Q(y), discharge given as a function of elevation '
    BCNOTE(3) = 'Q(y) = Conveyance*sqrt(S0),locally uniform flow  '
    BCNOTE(4) = 'y(t), elevation given as a function of time      '

    WRITE(OUTF,50)
50     FORMAT(64(1H=),/, 'THE RESULTS OF SIMULATION OF 1-D UNSTEADY FLOW',
1     ' IN SINGLE CHANNEL',/,64(1H=),/, 'TITLE OF THE RUN:')
    DO 10 I = 1, NTITLE
10     WRITE(OUTF, '4X,A') TITLE(I)
    WRITE(OUTF,100) FILENAME, GRAV, ALPHA, BETA, THETA,
1     TOTALMINUTES, DTMIN, NDT, NDX
100    FORMAT ('INPUT FILE:',/,4X,A,/, 'OUTPUT FILE:',/,4X,A,/,
1     'GENERAL DATA:',/,
2     4X,'Gravity acceleration =',F9.2,' m/sec2',/,
3     4X,'Alpha =',F9.2,/,
4     4X,'Beta =',F9.2,/,
5     4X,'Theta =',F9.2,/,
6     4X,'Simulation Time =',F9.2,' minutes',/,
7     4X,'Time step =',F9.2,' minutes',/,
8     4X,'Number of time step =',I6,/,
9     4X,'Number of points =',I6)
    WRITE(OUTF,110) LENGTH/1000.0, STRICKLER, SLOPE
110    FORMAT ('MAIN CHANNEL:',/,
1     4X,'Length =',F9.2,' km',/,
2     4X,'K Strikler =',F9.2,/,
3     4X,'Bottom Slope =',E9.2)
    IF (INITCONS.EQ.1) WRITE(OUTF,120) QINIT, YINIT
120    FORMAT ('CONSTANT INITIAL CONDITION:',/,
1     4X,'Initial Discharge =',F9.2,' cms',/,
2     4X,'Initial Elevation =',F9.2,' m')
    IF (STEADY) THEN
        WRITE(OUTF, 'A') 'NO STEADY STATE STABILIZATION'
    ELSE
        WRITE(OUTF,125) EPS, MAXCOUNT
    ENDIF
125    FORMAT ('STEADY STATE STABILIZATION:',/,
1     4X,'Eps =',E9.2,/,
2     4X,'Maximum step =',I9,' time steps')

```

```

WRITE(OUTF,200) IUSTYP, BCNOTE(IUSTYP), IDSTYP, BCNOTE(IDSTYP)
200  FORMAT ('BOUNDARY CONDITION:',/,4X,'U/S:',I2,'. ',A,/,
1      4X,'D/S:',I2,'. ',A)

IF (NIBC.NE.0) THEN
  WRITE (OUTF,'(A)') 'INTERNAL BOUNDARY CONDITION:'
  DO 300 I = 1, NIBC
    IF (IBCTYP(I).EQ.1) THEN
      XKM = X(IIBC(I))/1000.0
      WRITE (OUTF,210) IBCTYP(I), XKM, IIBC(I)
210   FORMAT(I5,': Inflow/outflow from outside at x = ',
1       F6.1, ' km (i =', I3,')')
    ELSE IF (IBCTYP(I).EQ.2) THEN
      XKM = X(IIBC(I))/1000.0
      WRITE (OUTF,220) IBCTYP(I), XKM, IIBC(I),
1       FLOORELEV
220   FORMAT(I5,': Lateral storage pocket at x = ',
1       F6.1, ' km (i =', I3,')',/,
2       7X, 'Floor Elev. =',F8.2,' m')
    ELSE IF (IBCTYP(I).EQ.3) THEN
      XKM = X(IIBC(I))/1000.0
      WRITE (OUTF,230) IBCTYP(I), XKM, IIBC(I),
1       CGATE, AGATE
230   FORMAT(I5,': Sluice gate at x = ',
1       F6.1, ' km (i =', I3,')',/,7X,
2       'Cd =',F8.2,/,7X,'Area =',F8.2, ' m2')
    ELSE IF (IBCTYP(I).EQ.4) THEN
      XKM = X(IIBC(I))/1000.0
      WRITE (OUTF,240) IBCTYP(I), XKM, IIBC(I),
1       CWEIR, BWEIR, YWEIR, HWEIR
240   FORMAT(I5,': Weir at x = ',
1       F6.1, ' km (i =', I3,')',/,7X,
2       'Disch.Coeff. =',F8.2,/,7X,
3       'Crest Width =',F8.2,' m',/,7X,
4       'Crest Elev. =',F8.2,' m',/,7X,
5       'Height =',F8.2,' m')
    ELSE
      XKM = X(IIBC(I))/1000.0
      WRITE (OUTF,250) IBCTYP(I), XKM, IIBC(I),
1       CINTAKE, BINTAKE, YINTAKE
250   FORMAT(I5,': Free flowing intake at x = ',
1       F6.1, ' km (i =', I3,')',/,7X,
2       'Disch.Coeff. =',F8.2,/,7X,
3       'Crest Width =',F8.2,' m',/,7X,
4       'Crest Elev. =',F8.2,' m')
  ENDIF
300  CONTINUE
ELSE
  WRITE (OUTF,'(A)') 'NO INTERNAL BOUNDARY CONDITION'
ENDIF

WRITE(OUTF,350) IFREQ, NPRINT, (IPRINT(I), I=1,NPRINT)
350  FORMAT ('OUTPUT CONTROLS:',/,
5      4X,'Frequency of printout =',I5,' time steps',/,
6      4X,'Number of Location =',I5,/,
7      4X,'At Location =',I5,10I4)

WRITE(OUTF,400) ('-----', I=1,NPRINT)
WRITE(*,401) ('-----', I=1,NDX,NDX-1)
400  FORMAT (64(1H=),///,'ELEVATION AND DISCHARGE HYDROGRAPHS',/,
1      '-----',10A)
401  FORMAT (///,' ELEVATION AND DISCHARGE HYDROGRAPHS',/,
1      '-----',10A)
WRITE(OUTF,500) (' Y(',IPRINT(I),') Q(',IPRINT(I),
1      ')',I=1,NPRINT)
WRITE(*,501) (' Y(',I,') Q(',I,
1      ')',I=1,NDX,NDX-1)
500  FORMAT ('Minute',20(A,I3,A,I3,A))
501  FORMAT (' Minute',20(A,I3,A,I3,A))
WRITE(OUTF,600) ('-----', I=1,NPRINT)

```

```

WRITE(*,601) ('-----', I=1,NDX,NDX-1)
600 FORMAT ('-----',10A)
601 FORMAT ('-----',10A)

RETURN
END

C0***6****1*****2*****3*****4*****5*****6*****77
C
C   These are collections of utilities needed in the Main Program.   C
C
C0***6****1*****2*****3*****4*****5*****6*****77
SUBROUTINE CHECKFILE (NOPT,FILENAME,notOK)
-----
CHARACTER FILENAME*20, BLANK*20, CHAR*1
LOGICAL notOK, ISEXIST

DATA BLANK/'          ','/', CHAR/' '/'

GO TO (10, 11) NOPT

C
C   -----
C   REQUEST FOR INPUT FILE
C   -----
10 WRITE(*,'(A)') ' Input data filename = ?'
READ(*,'(A20)') FILENAME
notOK = FILENAME.EQ.BLANK
IF (notOK) RETURN
INQUIRE(FILE=FILENAME,EXIST=ISEXIST)
IF (ISEXIST) RETURN
WRITE(*,'(A)') ' Your input file does not EXIST, try again please'
GO TO 10

C
C   -----
C   REQUEST FOR OUTPUT FILE
C   -----
11 WRITE(*,'(A)') ' Output data filename = ?'
READ(*,'(A20)') FILENAME
notOK = FILENAME.EQ.BLANK
IF (notOK) RETURN
INQUIRE(FILE=FILENAME,EXIST=ISEXIST)
IF (ISEXIST) THEN
12 WRITE(*,'(A)') ' Your output file already EXISTS'
WRITE(*,'(A)') ' Overwrite (Y/N) ?'
READ (*,'(A1)') CHAR
IF ( (CHAR.EQ.'N') .OR. (CHAR.EQ.'n') ) GO TO 11
IF ( (CHAR.NE.'Y') .AND. (CHAR.NE.'y') ) GO TO 12
ENDIF

99 RETURN
END

C0***6****1*****2*****3*****4*****5*****6*****77
SUBROUTINE OPENFILES (INF,OUTF,FILENAME)
-----
INTEGER INF,OUTF,theSTATUS
CHARACTER FILENAME(2)*20
LOGICAL notOK

C=====
C I. REQUEST FOR AN INPUT AND OUTPUT FILES
C=====
CALL CHECKFILE(1,FILENAME(1),notOK)
IF(notOK) STOP 'Good bye baby !'
OPEN (INF,FILE=FILENAME(1),IOSTAT=TheSTATUS)
IF (TheSTATUS.NE.0) THEN
WRITE(*,'(A)') ' *** I/O ERROR : ', TheSTATUS
STOP
ENDIF

```



```
CALL CHECKFILE(2,FILENAME(2),notOK)
IF (notOK) STOP 'Good bye baby !'
OPEN (OUTF,FILE=FILENAME(2),IOSTAT=TheSTATUS)
IF (TheSTATUS.NE.0) THEN
  WRITE(*,'(A)') ' *** I/O ERROR : ', TheSTATUS
  STOP
ENDIF

RETURN
END
```

```
C0***6****1*****2*****3*****4*****5*****6*****77
```

```
  SUBROUTINE RAZCONS(TABLE,LENGTH,CONS)
```

```
C-----
  DIMENSION TABLE(LENGTH)
```

```
100 DO 100 I = 1,LENGTH
    TABLE(I) = CONS
    RETURN
END
```

# STRUKTUR DATA MASUKAN

Last modified : Monday, December 16, 1991

INPUT RECORD	VARIABLE	FORMAT	DESCRIPTION
1	NTITLE	I	Number of lines that TITLE occupies, maximum 5 lines.
	TITLE	A60	60-character title of run in a line, total = NTITLE lines
2	GRAV	F	Gravity acceleration (m/sec <sup>2</sup> )
	ALPHA	F	Coriolis coefficient
	BETA	F	Space weighting factor
	THETA	F	Time weighting factor
	TOTIMEM	F	Total time to simulate (minutes)
	DTIMEM	F	$\Delta t$ (minutes)
	STEADY	L	= T (.TRUE.) if steady state stabilization is needed to get the correct initial condition. = F (.FALSE.) if steady state stabilization is NOT needed.
	EPS	F	Tolerance for steady state stabilization:
MAXCOUNT	I	Maximum number of iteration to get a steady state initial condition	
3	CONSDX	I	$\leq 0$ if $\Delta x$ is a constant. Followed by Record 3A = 1 if $\Delta x$ varies. Followed by Record 3B = 2 if $\Delta x$ semi-constant. Followed by Record 3C
3A	NDX	I	Number of computational grid points
	DX	F	$\Delta x$ in meter
3B	NDX	I	Number of computational grid points
	X	F	X-coordinate of computational grid points.of total NDX points
3C	NXREGION	I	Number of regions in which constant $\Delta x$ applies
	XUS	F	The most upstream x-coordinate (m).

Steady state stabilization will be terminated if tolerance is less than EPS or number of iterations exceed MAXCOUNT.

XDS	F	Downstream x-coordinate of a region
NREACH	F	Number of reaches in a region

XDS and NREACH must inputted in pairs for each NXREGION.

4	STRICKLER SLOPE	F F	Strickler coefficient Bottom slope
5	INITCONS	I	Flag for initial condition = 1, constant initial condition, and then followed by Record 5A = 2, non-constant initial condition, and then followed by Record 5B = 3, semi-constant initial condition, and then followed by Record 5C = 4, non-constant initial condition. User must supply a SUBROUTINE that defines the initial condition
5A	QINIT	F	Discharge for initial condition (m <sup>3</sup> /sec.)
	YINIT	F	Elevation for initial condition (m)
5B	Q	F	Discharge for initial condition (m <sup>3</sup> /sec.), one for each computational point. Total = NDX
	Y	F	Elevation for initial condition (m), one for each computational point. Total = NDX
5C	QINIT	F	Discharge for initial condition (m <sup>3</sup> /sec.)
	NYREGION	I	Number of regions in which semi-constant initial conditions apply
	YINITUS	F	The most upstream elevation for initial condition (m)
	YINITDS	F	Downstream water elevation of a region
	NREACH	I	Number of reaches in a region

**IMPORTANT:**

Record 5B must be inputted in pairs [Q(i), Y(i)], one for each computational point (i = 1 to NDX). Total = NDX.  
Record 5C, YINITDS and NREACH must be inputted in pairs for each NYREGION.

6	IFREQ	I	Frequency of output in $\Delta t$
---	-------	---	-----------------------------------

	NPRINT	I	Total number of stations to print
	IPRINT	I	The indices of the above stations
7	IUSTYP	I	Code number for U/S boundary condition 1: Q(t), Q hidrograp, followed by Record 7A 2: Q(y), "rating curve" followed by Record 7A 3: Q(y) = Conveyance * $\sqrt{S_0}$ , "locally uniform flow" 4: Y(t), Y hidrograp, followed by Record 7A
7A	NBCUS	I	Total number of datum in time data list, example : Time or Y            Y or Q x1            z1 x2            z2 x3            z3 x4            z4 then NBCUS = 4*2 = 8
	BCUS	F	The data list must be inputted in the following order: x1 z1 x2 z2 ..... xn zn.

**IMPORTANT:**

In the data list, in Record 7A, the unit must be correct:

Time always in minutes.

Surface elevation, Y, always in meter

Discharge, Q, always in m<sup>3</sup>/sec

8	IDSTYP	I	Code number for D/S boundary condition 1: Q(t), Q hidrograp, followed by Record 8A 2: Q(y), "rating curve" followed by Record 8A 3: Q(y) = Conveyance * $\sqrt{S_0}$ , "locally uniform flow" 4: Y(t), Y hidrograp, followed by Record 8A
8A	NBCDS	I	Total number of datum in time data list, example : Time or Y            Y or Q x1            z1 x2            z2 x3            z3 x4            z4 then NBCDS = 4*2 = 8
	BCDS	F	The data list must be inputted in the following order: x1 z1 x2 z2 ..... x4 z4.

**IMPORTANT:**

In the data list, in Record 8A, the unit must be correct:

Time always in minutes.

Surface elevation, Y, always in meter

Discharge, Q, always in m<sup>3</sup>/sec

9	NIBC	I	Total number of internal boundary conditions										
10	IIBC	I	Index of location of internal boundary condition										
	IBCTYP	I	Code number for internal boundary conditions: 1: Inflow/outflow from outside, followed by Record 10A 2: Lateral storage pocket, followed by Record 10B 3: Sluice gate, followed by Record 10C 4: Weir, followed by Record 10D 5: Freeflowing intake, followed by Record 10E										
10A	NINFLOW	I	Total number of datum in time data list, example : <table border="0" style="margin-left: 40px;"> <thead> <tr> <th>Time</th> <th>Discharge</th> </tr> </thead> <tbody> <tr> <td>t1</td> <td>Q1</td> </tr> <tr> <td>t2</td> <td>Q2</td> </tr> <tr> <td>t3</td> <td>Q3</td> </tr> <tr> <td>t4</td> <td>Q4</td> </tr> </tbody> </table> <p style="margin-left: 40px;">then NINFLOW= 4*2 = 8</p>	Time	Discharge	t1	Q1	t2	Q2	t3	Q3	t4	Q4
Time	Discharge												
t1	Q1												
t2	Q2												
t3	Q3												
t4	Q4												
	QINFLOW	F	The data list must be inputted in the following order: t1 Q1 t2 Q2 ..... t4 Q4. NOTE: t in minutes and Q in m <sup>3</sup> /sec										
10B	FLOORELEV	F	Elevation of the floor of storage pocket										
	NSURF	I	Total number of datum in time data list, example : <table border="0" style="margin-left: 40px;"> <thead> <tr> <th>Surface Elev.</th> <th>Surface Area</th> </tr> </thead> <tbody> <tr> <td>y1</td> <td>S1</td> </tr> <tr> <td>y2</td> <td>S2</td> </tr> <tr> <td>y3</td> <td>S3</td> </tr> <tr> <td>y4</td> <td>S4</td> </tr> </tbody> </table> <p style="margin-left: 40px;">then NSURF= 4*2 = 8</p>	Surface Elev.	Surface Area	y1	S1	y2	S2	y3	S3	y4	S4
Surface Elev.	Surface Area												
y1	S1												
y2	S2												
y3	S3												
y4	S4												
	SURF	F	The data list must be inputted in the following order: y1 S1 y2 S2 ..... y4 S4. NOTE: y in meter and S in m <sup>2</sup>										
10C	CGATE	F	Discharge coefficient of the gate										
	AGATE	F	The area of gate opening (m <sup>2</sup> )										

<b>10D</b>	CWEIR	F	Discharge coefficient of the weir
	BWEIR	F	Crest width of the weir (m)
	YWEIR	F	Crest elevation of the weir (m)
	HWEIR	F	Depth of the weir (m)
<b>10E</b>	CINTAKE	F	Discharge coefficient of the intake
	BINTAKE	F	Crest width of the intake(m)
	YINTAKE	F	Crest elevation of the intake(m)

One Record 10 for each of internal boundary condition.  
Total = NIBC.

## DAFTAR PUSTAKA

- Abbot, M.B., "Computational Hydraulics – Elements of the Theory of Free Surface Flows," Pitman Publishing Limited, 1979.
- Chow, V.T., Ph.D., "Open-Channel Hydraulics," McGraw-Hill Kogakusha, Ltd., International Student Edition, 1959.
- Cunge, J.A., F.M. Holly, Jr., and A. Verwey, "Practical Aspect of Computational River Hydraulics," Pitman Advanced Publishing Program, 1980.
- French, Richard H., "Open-Channel Hydraulics," McGraw-Hill Book Company, 1985.
- Henderson, F.M., "Open Channel Flow," Macmillan Publishing Co., Inc., 1966.
- "Computational Hydraulics," Course# 53:273, A lecture given by Prof. Forrest M. Holly Jr., Iowa Institute of Hydraulic Research, The University of Iowa, Iowa 52242, USA.
- "The Programmer's Companion," PRIME FORTRAN 77, Revision 18, Prime Computer, Inc., 1982.
- "Unsteady Flow in Open Channels," Volume 1, Water Resources Publications, Fort Collins, Colorado, USA, Editor: K. Mahmood dan V. Yevjevich, 1975.