

下水研モデルプログラムマニュアル
(Ver.1.1.0 改訂版)

平成 24 年 12 月

国土交通省国土技術政策総合研究所下水道研究室

はじめに

下水研モデル及びそれを用いた解析ソフト「JAMSOOP」は、主として合流式下水道改善対策のための雨天時汚濁解析に用いられるソフトである。

この度、「JAMSOOP」の改良を行ったため、それに伴い、下水研モデルプログラムマニュアルも改訂を行った。それぞれの主な改良点を以下に示す。

(1) JAMSOOP

- ・修正 RRL 法を組み入れ、雨天時における路面流出量及び管内流量を JAMSOOP 内部で計算することが可能となった。
- ・雨水貯留施設の貯留容量の設定が可能となった。
- ・下水処理場における流量及び汚濁負荷量の計算が可能となった。
- ・複数降雨を対象とした年間解析・集計機能が可能となった。これにより、処理場や雨水貯留施設等を対象とした年間の総汚濁負荷量計算や合流式下水道改善対策施設による効果を計算することが可能となった。

(2) 下水研モデル

- ・従来の土研モデルの考えを踏まえ、マンホールへ流入する地表面負荷量が、マンホール直下の管路の上流端では管内堆積負荷量に付加されず、マンホール直下の管路の下流端（もしくは下流側の接続管路の上流端）で管内由来の流出負荷に合算されるように変更した。

（マンホール直下の管路では、地表面からの負荷量と管内堆積負荷量はそれぞれ独立している。流域中流部に小さな面積の流域があると、上流からの汚濁負荷供給が過大な負荷となり、収束計算が不安定になる場合があるため、上記の通り変更した。）

目 次

1. 総説	1-1
1-1. 分布型モデル(下水研モデル)開発の背景と目的	1-1
1-2. 雨水流出モデルと負荷流出モデルの関係	1-6
1-3. 下水研モデルの特徴	1-7
2. 基礎式と式の展開	2-1
2-1. 汚濁負荷解析のモデル化の考え方	2-1
2-2. モデルの基礎式と解説	2-4
3. モデル解析作業	3-1
3-1. 解析のプロセス	3-1
3-2. 入力項目(境界条件)	3-3
3-3. 計算手順	3-12
3-4. 出力項目	3-13
3-5. プログラムの制限事項	3-15
4. パラメータの考え方(設定方法)	4-1
4-1. 管路パラメータの設定方法	4-2
4-2. 地表面パラメータ設定に関する参考資料	4-5
5. ケーススタディ	5-1
5-1. 仮想モデル地区	5-1
5-2. シミュレーション結果	5-4
5-3. 浸透施設の整備	5-10
5-4. 滞水池の整備	5-17
5-5. 分流化	5-23
5-6. 各ケースの比較	5-31
6. プログラム解説とプログラムリスト	6-1
6-1. 主要変数及び各ルーチン	6-1
6-2. プログラムリスト	6-8
参考文献	参-1

1. 総説

1. 総説

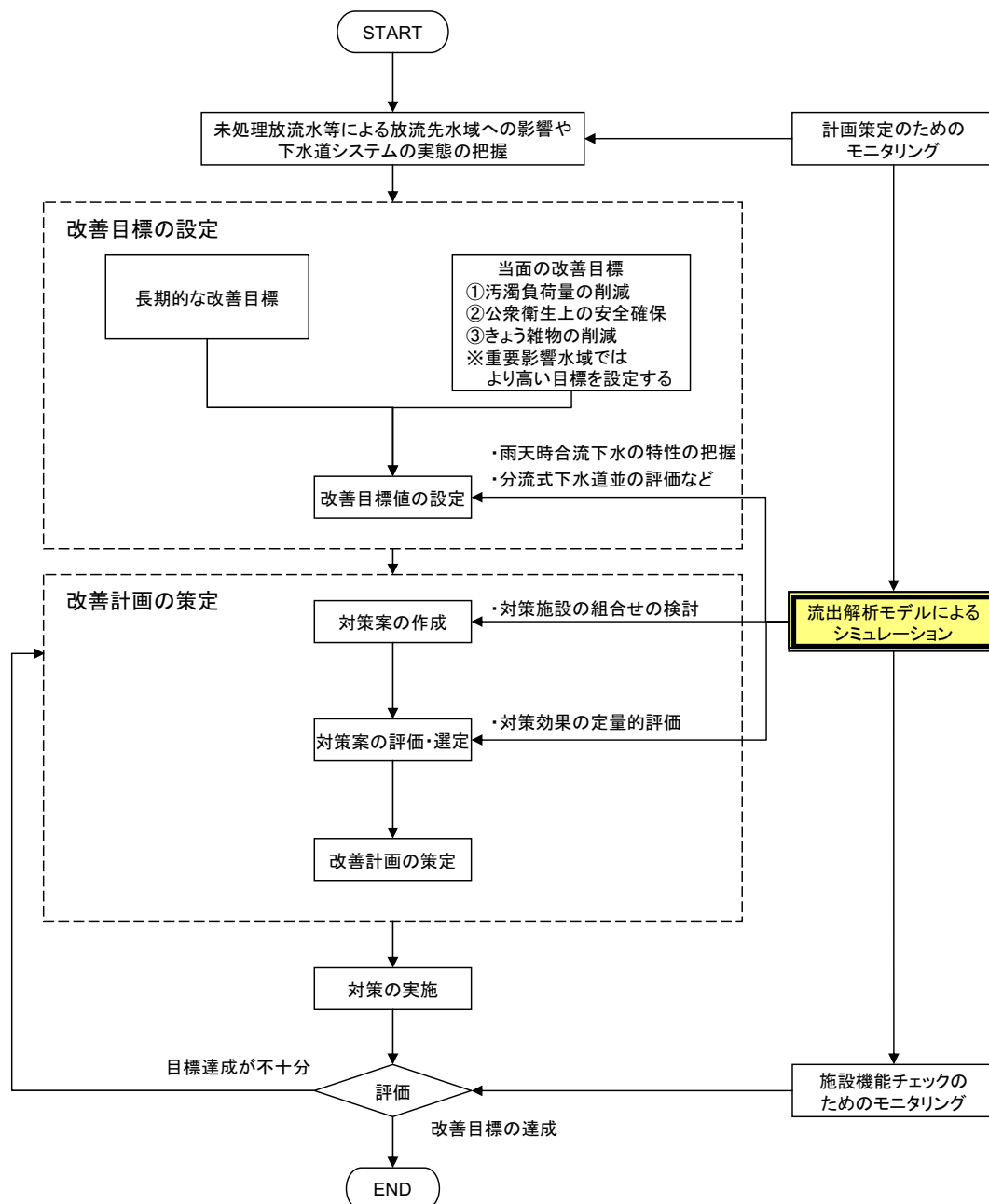
1-1. 分布型モデル（下水研モデル）開発の背景と目的

従来の集中型流出解析モデル(土研モデル)には流末における評価しかできないというモデル化の限界があり、また市販の分布型流出解析モデルにはモデル定数の検証やライセンス料が高価である、といった課題がある。これらを受け、無償公開の分布型汚濁負荷流出解析モデルである**下水研モデル**の開発を行った。

- (1) 集中型モデルの課題
- (2) 市販されている分布型モデルの課題
- (3) 国産分布型モデル（下水研モデル）開発の必要性

(1) 集中型モデルの課題

合流式下水道計画を策定する際、「雨天時下水の特性把握」、「対策施設の定量的評価」等を行うため、「流出解析モデルによるシミュレーション」は不可欠となる（図 1-1）。



注1: 流出解析モデルによるシミュレーションは、改善目標値の設定と改善計画策定において一体的に行うことが可能。

注2: 既に合流改善計画を策定あるいは実施している都市においては、既計画を十分に評価し活用すること。

図 1-1 合流改善計画策定フロー¹⁾

日本では、合流改善等の検討する際の雨水・負荷量モデルとして、修正 RRL 法および土研モデルが広く用いられてきた。しかし、同モデルには以下に示すような課題がある。

①モデル化の限界

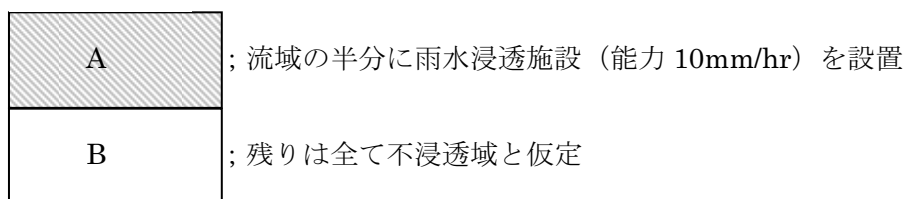
合流改善の場合、対策効果の評価は吐き口や処理場放流量など原則として流末で行われることが多かった。また具体的な対策も、遮集倍率のアップや雨水滞水池の設置など、流末段階で組み込めばよいものがあった。

しかし近年は、雨水貯留浸透などの面的に実施される施策がクローズアップされてきており、集中型モデルではこれら施策を的確に評価できないケースも考えられる。

例えば部分的に雨水浸透を実施した場合、現実には、施設を設置した箇所ではしか効果を発揮できない。しかし、集中型モデルの場合は原則として流域平均でしか評価できないため、小降雨の場合は計算上流域全体が 100% 効果を発揮してしまう場合がある（下記検討例参照）。

以上のように、集中型モデルは一般に大きな流域単位でモデル化を行うため（きめ細かな対応が困難）、種々の対策を必ずしも適切にモデル化できるとは限らない。

【検討例】



このような流域に 5mm/hr の降雨があった場合、A からは全く流出しないが B からは 100% 流出する。

一方、これを集中型モデルで表現した場合、一般には流域平均の浸透能力を 5mm/hr で与えるため、降雨 5mm/hr は全く流出しないことになる（流域全体としての流出量がゼロとなる）。

雨水滞水池の設置や増補管渠の新設など、流域内に新たな施策を展開する場合は、流域分割を行うなど擬似的な分布型モデルを構築し検討を進めているケースが見られる。

しかし、対策位置に変更が生じた場合などは、流域分割の見直しが生じ作業が煩雑になるとともに、モデル定数の変化に伴い計算値にも変化が生じ（流域流出量の単純合計が必ずしもケース毎に一致しない）、モデル化の妥当性に課題が残る。

(2) 市販されている分布型モデルの課題

これに対し、海外で開発された InfoWorks、MOUSE、XP-SWMM といった分布型の雨水・負荷量モデルを使った検討が近年進められつつある²⁾。

しかし、これらの海外モデルの使用に当っては、以下に示すような課題がある。

①モデル定数の検証

集中型モデルである修正 RRL 法+山口・中村モデル³⁾については、国内で過去数十年間に渡る多くの検討実績があり、その定数についても水質調査結果に基づいた設定方法や値の目安が示されている。このため、検討の際には適切な定数の設定が可能であると考えられる。

しかし、市販されている各モデルによる国内での検討実績は未だ少なく、またモデルにより水質推定の理論が異なることもあり、各モデルで利用されている定数の国内での適用性については、十分な検証が行われていない状況である。

②プログラムの修正が困難

プログラムの修正はソフトの開発元しかできない。

このため、新たな技術や既存のモデルでは表現できない施策を展開する必要性が生じた場合でも、迅速に対応できない可能性がある。

またコマーシャルベースのため、利益が期待できない改良には、依頼者側に多大な経費が発生する可能性がある。

③ソフトが高価である

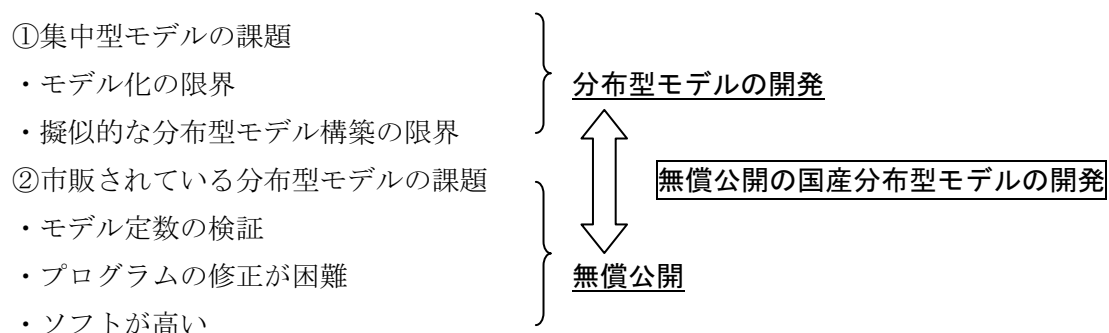
ソフトにより多少の違いはあるが、総じてソフト代が高く、購入代金は1ライセンスあたり数百万円、保守費用は年間あたり購入代金の1割程度の場合が多い。

このため、誰でもが気軽に使えるという状況にはない。

(3) 国産分布型モデル開発の必要性

このような現状を踏まえ、無償公開を前提とした国産の分布型モデルの開発が有効であると考えられる。

このことから、本検討では現状広く用いられている土研モデルを改良し、分布型汚濁負荷解析モデルである下水研モデル (WSD Model)の開発を行った。



下水研モデルの特徴、基本原理、解析方法や解析事例等について以下に説明する。

1-2. 雨水流出モデルと負荷流出モデルの関係

下水研モデルは汚濁負荷量の流出解析を行うモデルであり、雨天時流量については、修正 RRL 法を分布型に拡張したモデルを使用する。

従来型の解析の場合、雨水流出解析は修正 RRL 法で行い、負荷量解析は土研モデルを適用していた⁴⁾。

分布型汚濁負荷の解析を行う場合、モデル化した管路施設全てに対して、時系列の流量データが必要となる。したがって、雨水流出解析モデル自体、分布型を用いる必要があり、従来から使用されていた修正 RRL 法では不十分である。

したがって雨水流出解析については、各小流域について従来の修正 RRL 法を適用し、流下時間および管網の接続状況に応じて流下させて計算するよう、分布型に拡張した修正 RRL 法を導入した。

なおその他の分布型雨水流出解析モデルとしては、現在以下に示すものがある。

日本国内では旧建設省土木研究所都市河川研究室で開発された PWRI モデル(現 NILIM モデル)⁵⁾がある。

NILIM モデルは氾濫解析を目的としたモデルであり、一次元および二次元の氾濫解析が可能である。

但し、浸水等を引き起こす大規模な降雨を主な解析対象としていることから、合流式下水道の越流水対策の場合に解析対象とする中小規模の降雨への適用には注意が必要である。

その他の分布型雨水流出解析モデルとしては、海外で作成された HydroWorks (現 InfoWorks)、MOUSE、XP-SWMM 等がある²⁾。

1-3. 下水研モデルの特徴

今回開発した下水研モデルは分布型汚濁負荷解析モデルであり、水質 5 項目(BOD、COD、SS、T-N、T-P)の解析が可能である。無償でソースコードを公開するため経済的であり、ユーザによる改良が容易である一方、解析条件や操作性において更なる課題を残すものである。

- (1) モデルの名称等
- (2) 適用範囲と設定条件
- (3) 下水研モデルの優位性
- (4) 下水研モデルの課題

(1) モデルの名称等

本モデルの名称は、開発元である国土交通省国土技術政策総合研究所下水道研究室の略称「下水研（ゲスイケン）」に因み、**下水研モデル**とした。また、下水道研究室の英語名（Wastewater System Division）の頭文字から、英語では WSD Model という略称を用いることとする。WSD Model は、Wastewater load Simulation Distributed Model の意味も兼ねている。

モデルの名称については、これまでの「土研モデル」を改良して開発された経緯から「分布型土研モデル」とすることも考えられたが、「集中型土研モデル」との区別がつきにくいこと、土研モデルの開発後に開発元である国総研が設立されており、土研の名前を継承して用いることは開発者が分かりにくいこと等の問題があるため、上記のとおり定めたものである。

(2) 適用範囲と設定条件

① 対象水質項目

下水研モデルでは、BOD, COD, SS, T-P, T-N の 5 項目について、分布型汚濁負荷解析が可能である。

大腸菌群数および糞便性大腸菌群数については、モデル化に必要なパラメータについて現時点で十分な検討がなされていないため対象外とした。

② 管路名(または小流域名)の付け方

計算は上流側から行い、計算する順序の早い管路(または小流域)の名称から、1,2,3,4...と連番にする。分岐する管路については、下記「分水機能について」を参照。

③ 放流・バイパス機能

流域内に放流やバイパス管が存在する場合を考慮し、遮集倍率やバイパス先管渠を指定することでこれらを表現する設定とした。

(3) 下水研モデルの優位性

開発した下水研モデルを無償公開する事により、以下のメリットと発展性が期待される。

① 技術的メリット

- 各種施策等を適切にモデル化できる可能性が高い
原則として流域情報を含む個々の管渠情報そのものをモデル情報として与えるため、面的な施策や流域途中で展開される施策をそのままモデル上に展開することが可能であり、集中型モデルに比較しその施策を適切に反映できる可能性が高い。
- モデル変更が容易
分布型モデルでは、入力データそのものが細分化されるため、例えば対策位置の変更等、対策計画の変更や将来計画に合わせたモデルの修正が相対的に容易となる。
- モデル定数設定に際して既往の知見を活用できる
今回開発したモデルは、既存の集中型モデルである山口・中村モデル³⁾をベースにしているため、モデル定数の意味合いが原則として同じ扱いとなっている。このため、モデル定数設定に際しては、既往の知見を活用することが可能である。
- プログラムの修正が容易
原則としてプログラムのソースコードを公開するため、必要に応じて利用者側でプログラム修正を臨機応変に行うことができる。

② 経済的メリット

- ソフト購入費用がかからない
原則として無償公開としているため、ソフト購入費用がかからない。このため、必要なユーザがいつでも利用することができる。
- 海外モデルとデータの共有が可能
近年、浸水対策については分布型モデルで検討することが多くなってきている。本モデルで利用する各種情報は原則として他の分布型モデルの情報が活用でき、労力の省力化が可能と考えられる。
- 雨水流出モデルのみの活用も可能
従来の下水研モデルでは汚濁負荷解析のみ実施可能で、雨水流出解析は他の分布型モデルによる解析結果を入力ファイルとする必要があったため、今回のバージョンアップの際、分布型に拡張した修正 RRL 法による雨水流出解析機能を追加した。
これにより、雨水流出解析のみを行うことも可能となった。

③ 発展性

- ・ モデルのバージョンアップ

原則としてプログラムのソースコードを公開するため、プログラムの管理環境を整えることで、外部ユーザによるプログラムの改良や修正などの発展が期待できる。

- ・ モデルが軽く維持管理・運転管理にも容易に活用できる

既存の集中型モデルである山口・中村モデルをベースにしているため、演算時間等において軽量なものとなっている。このため、RTC（リアルタイムコントロール）など演算時間の制約が大きな課題に対しても、本モデルの適用可能性は高いものと考えられる。また、処理場や合流改善施設の運転管理といった面での活用も期待される。

（４）下水研モデルの課題

開発された下水研モデルの課題として、以下の点が挙げられる。

① 解析上の制限事項

- ・ 既往の知見が活用できる、演算時間が短い等のメリットとある意味では裏腹の関係にあるが、既存の集中型モデルである山口・中村モデル⁶⁾をベースにしており、このモデルは定数の数が少ないため、全ての現象を同一の定数で表現することは困難である。
- ・ 現在の管渠モデルは完全混合型モデルであり、同条件が適用できない環境下では、実現象とモデルの解析結果が合わなくなる可能性がある。
- ・ 下水研モデルの基本的原理として個別の流域における汚濁負荷パラメータ設定が可能であるが、今回開発されたプログラムでは、モデルの簡略化のため全流域で単一の汚濁負荷パラメータを適用する設定となっている(詳細は「3-2. 入力項目(境界条件)」および「3-5. プログラムの制限事項」を参照)。

これらの制限事項については、プログラムのソースコードの改良によって対応が可能と考えられる。今後の改良に期待したい。

② モデルの操作性

今回のプログラムのバージョンアップに併せ、インターフェースソフト「JAMSOOP (Japanese Analysis Model for Sewer Overflow with Open Program)」(図 1-2、図 1-3)の改良も行った。

本システムにより CSV ファイルからの読み込みや、データの直接入力および貼り付けなどが可能であり、合流式下水道改善計画の検討に最低限必要と考えられる機能を網羅させたものであるが、対象流域の平面図表示等の機能は持っておらず、入出力操作において市販の分布型モデル程度の操作性は有していない。



図 1-2 下水研モデル解析インターフェース「JAMSOOP ver.1.1.0」
(検討ケース選択画面例)

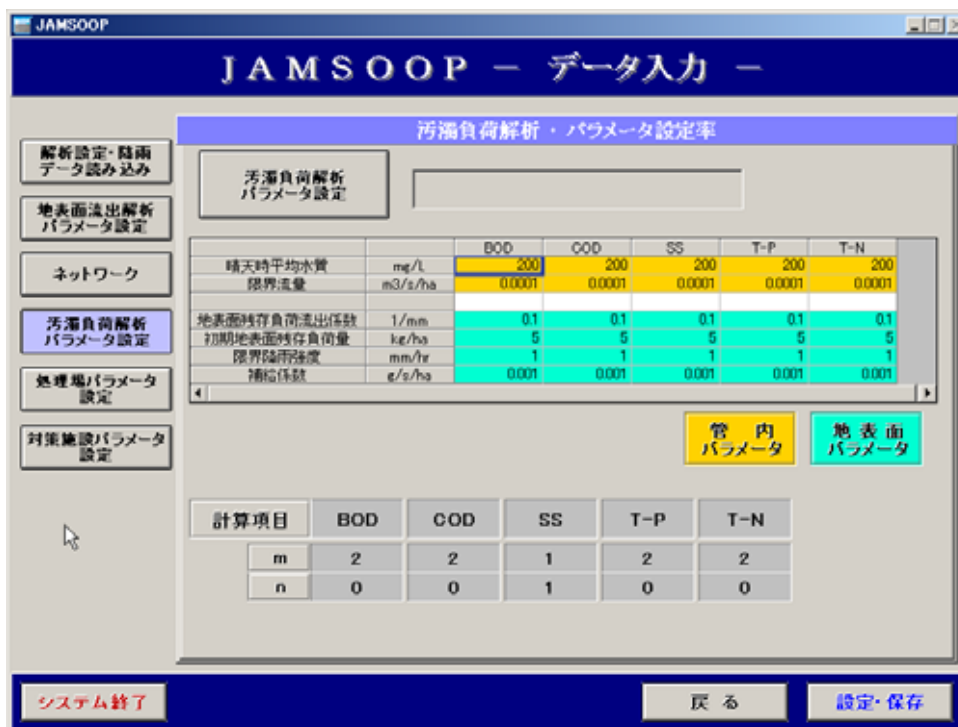


図 1-3 下水研モデル解析インターフェース「JAMSOOP ver.1.1.0」
(データ入力画面例)

2. 基礎式と式の展開

2. 基礎式と式の展開

2-1. モデルの考え方

今回のバージョンアップにより追加された流量解析モデルには、流域モデルとして修正 RRL 法を使用し、下流へは流下時間によるタイムステップのずれを考慮して受け渡すモデルとした。

また下水研モデルは、流域モデルと管路モデルに分かれており、流域モデルは従来の集中型モデル(土研モデル)の考え方を拡張して作成した。一方、管路モデルは流量解析結果と流域および上流から流入する汚濁負荷を用い、完全混合で流出する形式とした。

本検討で開発した分布型モデル(下水研モデル)は、図 2-1-1 に示すように「流域モデル」と「管路モデル」による基本ユニットのネットワークで構成される。

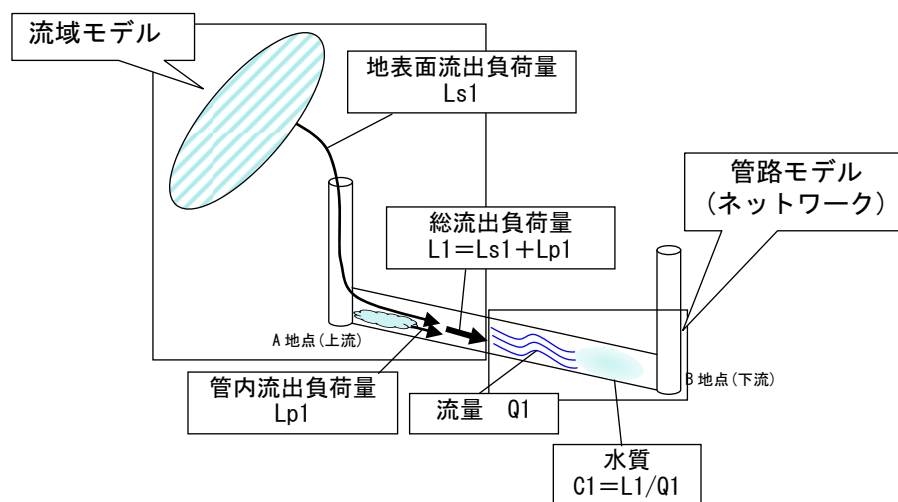


図 2-1-1 基本ユニット

下水研モデルは従来型の土研モデル⁴⁾を拡張し、分布型としたものである。したがって、基本的には修正 RRL 法+下水研モデルで算出した各小流域での流量および負荷量を積み上げたものが、従来型(集中型)での流量および負荷量となると考えることができる(図 2-1-2)。

下水研モデルにおける上流端の評価点は、従来型での評価と同一となると考えられることから、各評価点での評価は従来型のモデルで表現することが妥当と考えられる。

したがって、モデル化した管路施設への流量および汚濁負荷量の供給は、従来の集中型で表現することとした(流域モデル)。

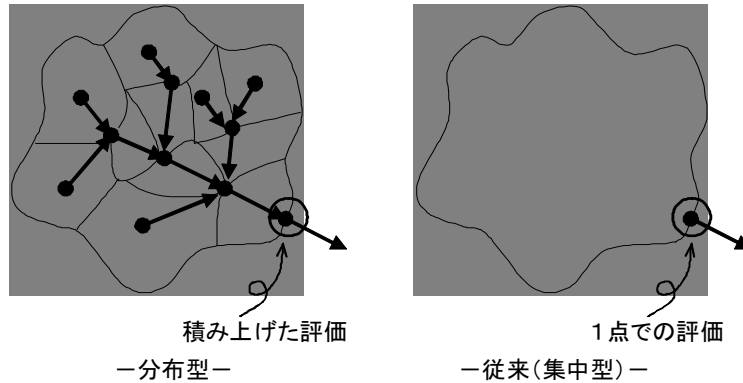


図 2-1-2 従来型と分布型の評価地点の違い

なお、バージョンアップ前の下水研モデルでは、流量および管内の貯留水量を外部からの入力値とし、地表面残存負荷量(P_s)の流入が管内堆積負荷量(P_p)へ寄与すると考える「供給型」としていたが、バージョンアップ後の下水研モデルでは流量についても修正 RRL 法によりモデル内で算出することとしている。

これに伴い、流域中流部に小さな面積の流域があると、上流からの汚濁負荷供給が過大な負荷となり、収束計算が不安定になる場合があることが判明した。

これを防ぐため、今回のバージョンアップに合わせ、従来の土研モデルをベースとして以下の計算方針とした。

- ・ 管内堆積負荷量と地表面残存負荷量をそれぞれ独立して計算
- ・ 地表面残存負荷は各流域で算出
- ・ 管内堆積負荷は各流域の上流域の堆積負荷を全て合算し、これと管内流量から流出負荷を算出
- ・ 上流域で雨水吐きやバイパスがある場合は、上記の堆積負荷合算値から雨水吐き等で流出する負荷を除外

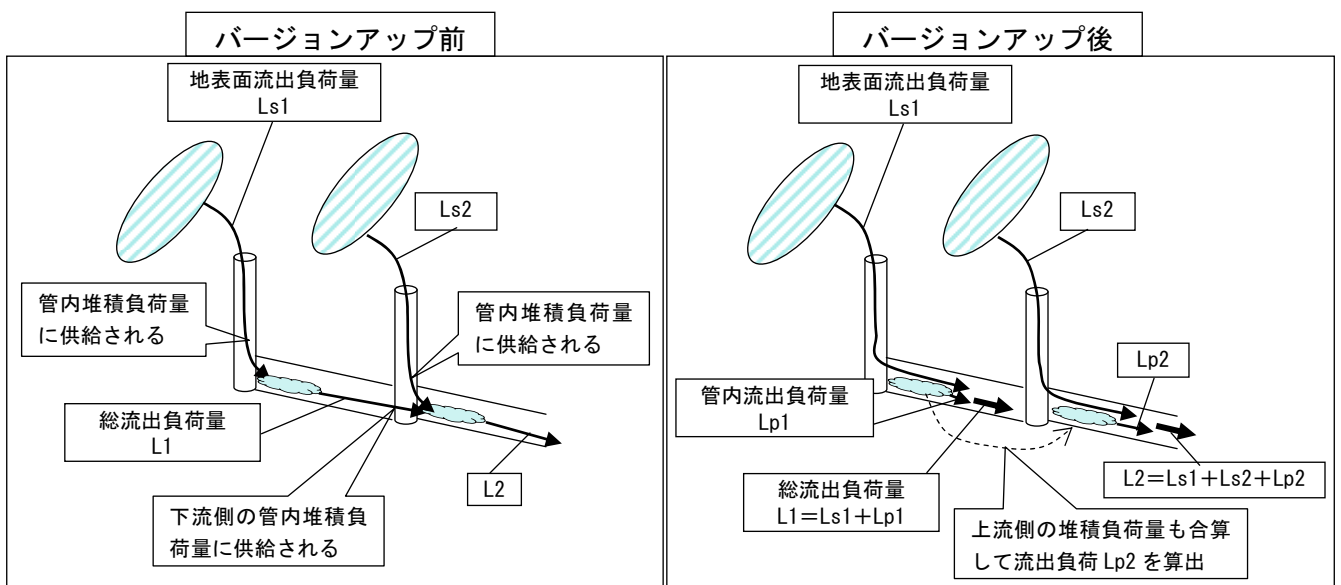


図 2-1-3 バージョンアップ前後の汚濁負荷解析の考え方の違い

「汚濁負荷供給は人孔部での流入のみ」と仮定すると、上流人孔から流入した汚濁負荷は、管路を伝い下流人孔へと移動し、そこで次の汚濁負荷の供給を受けることとなる。

モデル化する下水道管網全体からみて、1 管路の延長は微小のものと考えられることから、管路内の流量および汚濁負荷の移動は完全混合(水質は各管内で均一)とし、流下時間によるタイムステップのずれを考慮して下流に受け渡すこととした(管路モデル)。この場合、パラメータを必要としないというメリットもある。

なお、実際の分布型モデルを構築する際、主要な管路のみをモデル化(幹線モデル)することが想定され、その場合、モデル化されない管路施設は流域モデルとして考慮されることとなる。

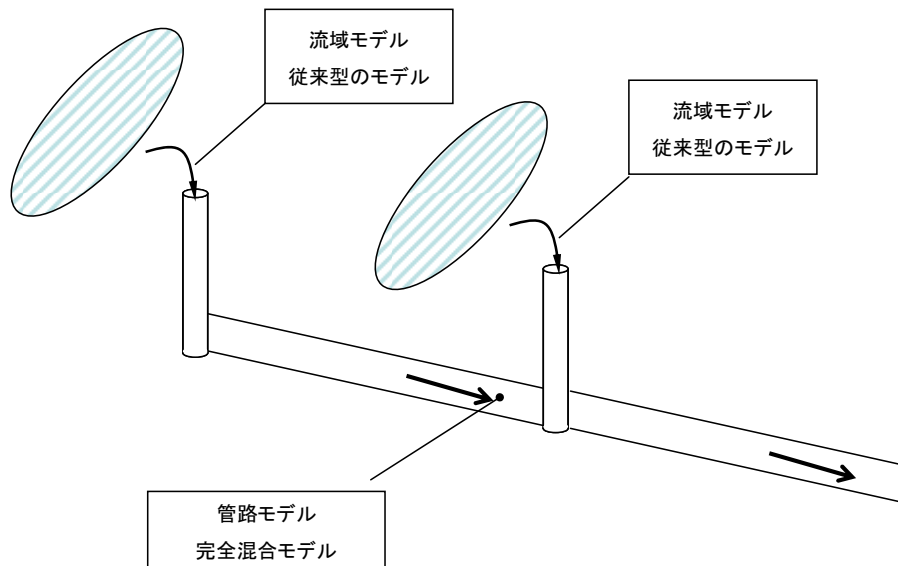


図 2-1-3 管路内の汚濁負荷の移動

2-2. モデルの基礎式と解説

下水研モデルは、上述のとおり流域モデルと管路モデルの二つによる基本構成となっている。下水研モデルによる汚濁負荷解析の事前に行う流量解析モデル(修正 RRL 法)も含め、各モデルの考え方および使用する基礎式について以下に解説する。

- (1) 流量解析モデル
- (2) 汚濁負荷解析モデル
- (3) 管内水質モデル

(1) 流量解析モデル

流量解析モデルは従来の修正 RRL 法を分布型に拡張したモデルを用いる。

修正 RRL 法の計算プロセスについて以下に示す。

修正 RRL 法は、以下のフローにより降雨量から流末流量を求める方法である。

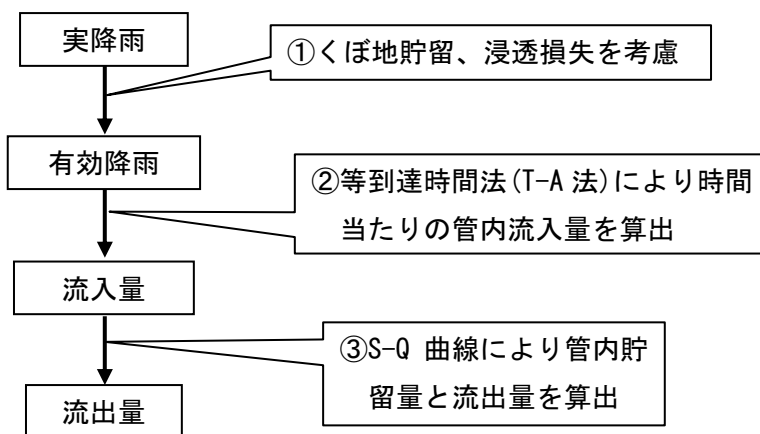


図 2-2-1 修正 RRL 法のフロー

1) 有効降雨の算出

流域に降った雨は全てが流出するわけではなく、公園や庭などの浸透域に降った雨は一部地下へ浸透し、更に水たまりにおいては一時的な貯留が生じ(凹地貯留)、そして浸透することになる。また不浸透域においても、水たまりが生じ貯留される。

修正 RRL 法では、このような都市域での降雨損失機構を、概ねそのままモデル化し、図 5-2-2 に示す様な有効降雨モデルを設定している。

すなわち、流域全体の平均有効降雨量(有効ハイトグラフ)は、4 種の各有効降雨量の面積加重平均により算定される。

①不浸透域の直接流出域

全量流出とする。

②不浸透域の凹地貯留域

不浸透域の凹地に貯留した雨水は、降雨後の蒸発や車にはねられないと損失しないため、降雨期間中は回復しないものとする。

③浸透域の直接流出域

浸透能は、一般に雨の降り始めに大きく、後に小さくなるが、修正 RRL 法では簡便のため一定として扱う。

④浸透域の凹地貯留域

浸透域の場合は、凹地に貯留した雨水も徐々に浸透する為、降雨強度が浸透能以下となれば貯留能力は回復する。

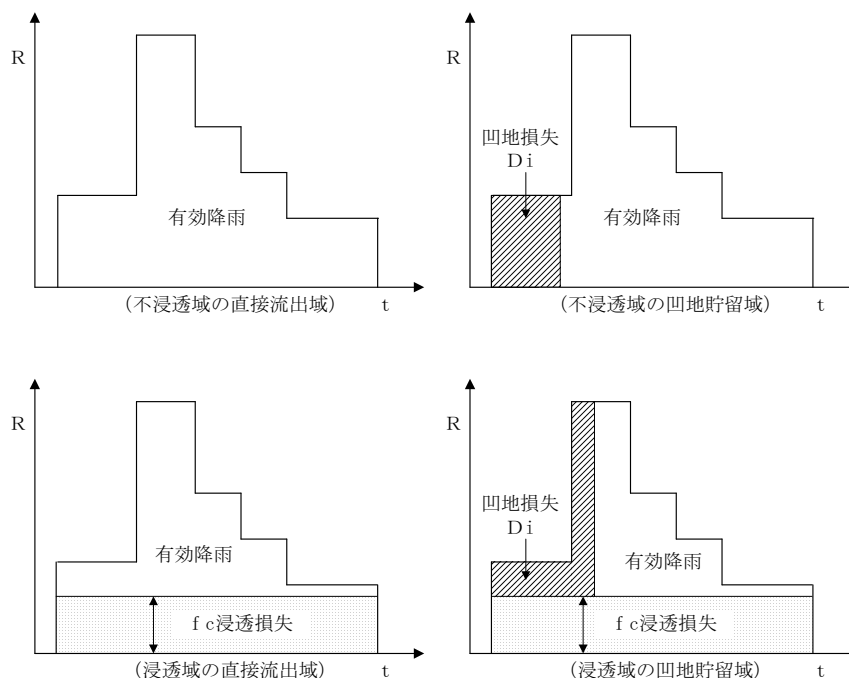


図 2-2-2 地表面工種と降雨損失の考え方

くぼ地貯留 D_i および浸透損失 F_c を考慮した有効降雨の算出方法は以下のとおりである。

・不浸透域の場合

$$D(t) = \text{Min}(D(t-1) + R(t), D_i) \dots\dots\dots \text{(式 2-2-1)}$$

$$\text{If } D(t) = D_i \text{ then } R_r(t) = D_i(t-1) + R(t) - D_i \dots\dots\dots \text{(式 2-2-2)}$$

$$\text{Else } R_r(t) = 0 \dots\dots\dots \text{(式 2-2-3)}$$

・浸透域の場合

$$D(t) = \text{Min}(D(t-1) + R(t) - f_c, D_i) \dots\dots\dots \text{(式 2-2-4)}$$

$$\text{If } D(t) = D_i \text{ then } R_r(t) = D(t-1) + R(t) - F_c - D_i \dots\dots\dots \text{(式 2-2-5)}$$

$$\text{Else } R_r(t) = 0 \dots\dots\dots \text{(式 2-2-6)}$$

ここに

- $R(t)$: 実降雨
- $R_r(t)$: 有効降雨
- $D(t)$: ステップ毎のくぼ地貯留量
- D_i : くぼ地損失
- F_c : 浸透損失

2) 管内流入量の算出

流域内に降った雨は、全て一定の時間で出口まで到達するのではなく、上流へ行くほど少しずつ時間ずれが生じる。修正 RRL 法では、時間ずれを考慮し流量を合成するために、5～10 分毎の等到達時間域を設定している。

等到達時間は、すべての管渠について、満管時(矩形管の場合は 9 割水深時)の流速で管の長さを割ったものを管網に従って累加計算することにより算定される。

円形管の場合について、マンニング式で計算するとすれば、懸案地点から k 番目の管渠の到達時間は次式で表わされる。

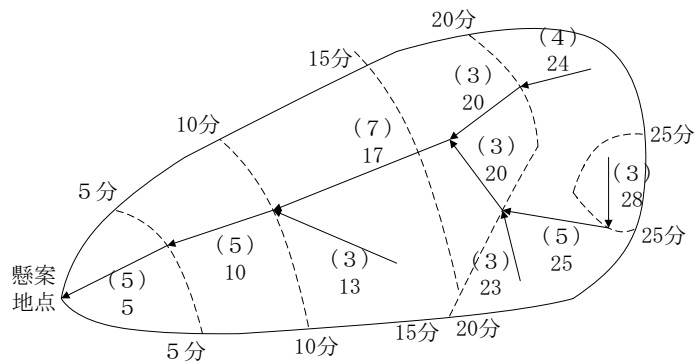
$$T_k = \sum_{i=1}^k t_i = \sum_{i=1}^k \frac{n_i \cdot L_i}{(D_i/4)^{2/3} \cdot I_i^{1/2}} \dots\dots\dots (式 2-2-7)$$

- ここに、 T = 到達時間
- t = 流下時間
- L = 管渠の延長
- D = 管渠の直径
- I = 管渠の勾配
- n = マニングの粗度係数

上記の計算から、図 2-2-3 に示す様な等到達時間域図が作成され、各到達時間に対する流域面積が算定される。

一方、流入ハイドログラフとは、等到達時間域毎に有効雨量を求め、それを単位時間毎にずらしながら合成したもので、S～Q 曲線により流量低減をさせて流出ハイドログラフを作成する前の仮想の流量波形である。

図 2-2-4 の例を使って具体的に説明する。I₁ は、降雨 R₁ による A₁ 区域からの流出量であり、I₂ は、R₁ による 1 つ上流の A₂ 区域から流出量と、次の時間の降雨 R₂ による A₁ 区域からの流出量の合計で、これを順時降雨時間数だけ累加したものが流入ハイドログラフとなる。



注) ()は個々の流下時間。裸字は累加時間。

図 2-2-3 等到達時間域図

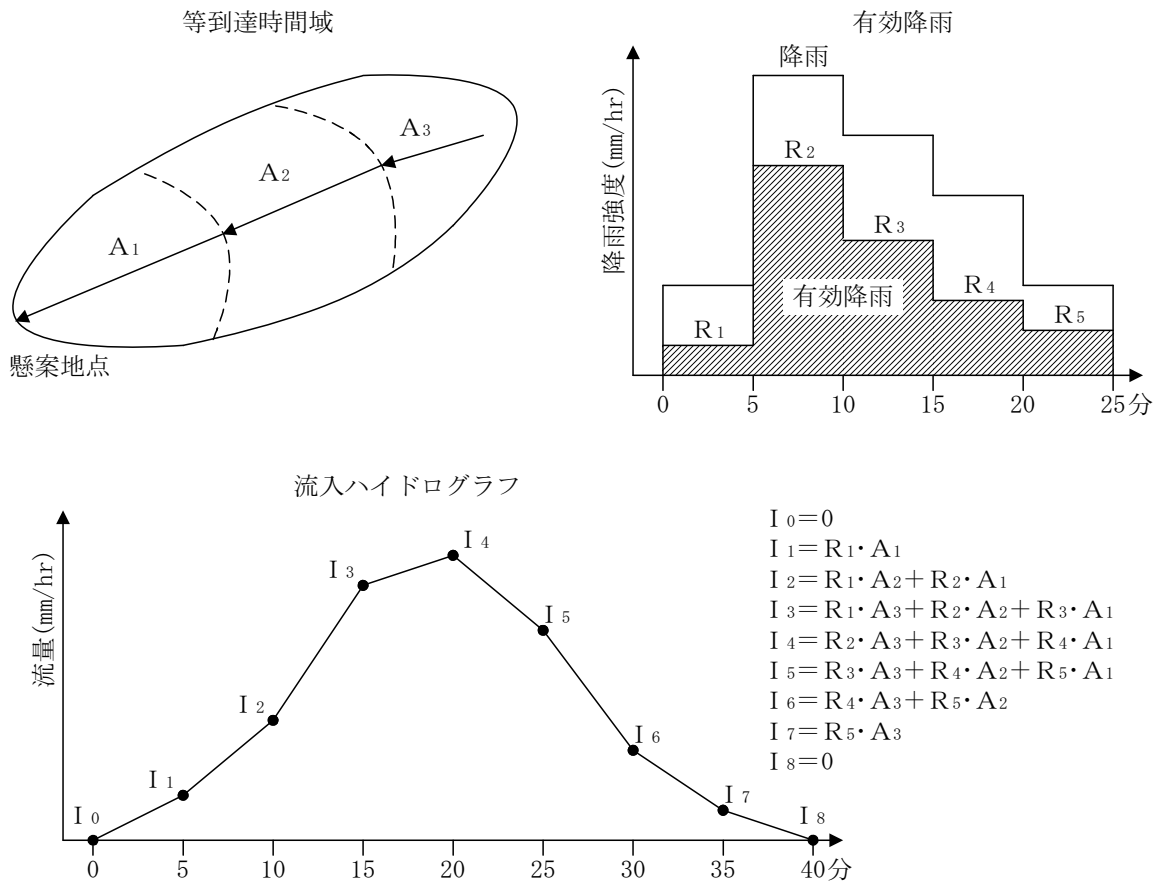


図 2-2-4 流入ヒドログラフの算定方法

3) 流出量の算出

修正 RRL 法では、流域に一定の強度の雨が降り続いた状態での各下水管渠の流れを等流と仮定し貯留量を算定し、その時の懸案地点での流出量Qと、累加した管内貯留量Sとの関係により、S～Qカーブを設定している。

$$Q = f \cdot R \dots\dots\dots (式 2-2-8)$$

$$S = \left(\sum a_i \cdot L_i \right) / A / 10 \dots\dots\dots (式 2-2-9)$$

- ここに、
- Q : 懸案地点の流出量 (mm/hr)
 - f : 流出係数
 - R : 降雨強度 (mm/hr)
 - S : 貯留量 (mm)
 - A : 流域面積 (ha)
 - a_i : 流下断面積 (mm) (等流計算により流出量 Q_i を変換)
 - L : 管渠長 (m)

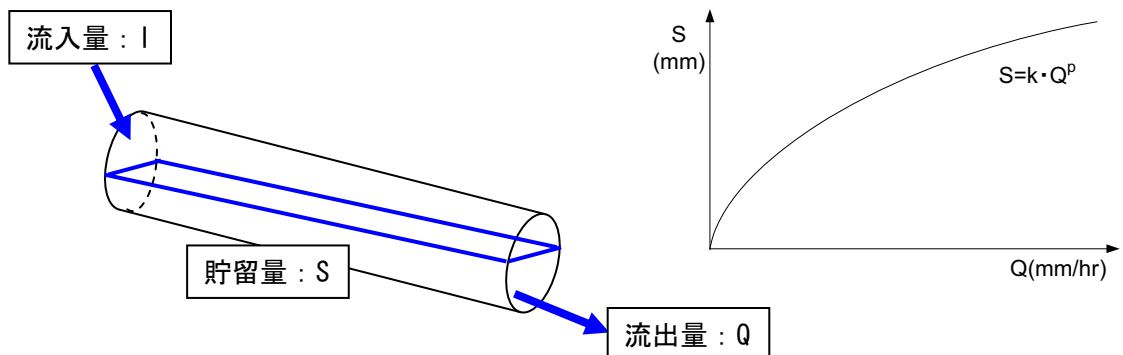
流出ハイドログラフは、流入ハイドログラフを流入量 I とし、S～Qカーブを用いて管内貯留による流量低減計算を行い、流出量Qを求めることにより算定される。

連続の式：
$$\frac{d_s}{d_t} = I - Q \dots\dots\dots (式 2-2-10)$$

差分式：
$$\frac{S_2 - S_1}{\Delta t} = \frac{I_1 + I_2}{2} - \frac{Q_1 + Q_2}{2} \dots\dots\dots (式 2-2-11)$$

$$\frac{Q_2 + S_2}{2} = \frac{I_1 + I_2}{2} - \frac{Q_1 + S_1}{2} \dots\dots\dots (式 2-2-12)$$

未知量 既知量(第 1step 目の Q_1, S_1 は 0 とする)



(流入量 I は T-A 法より求められる)

図 2-2-5 S-Q 曲線の概念

(2) 汚濁負荷解析モデル

① 基礎式

流域モデルの汚濁負荷量を算定する基本式は以下の式に従う。なお、2-1 項での説明のとおり、収束計算の安定化のため、従来の土研モデルの式を採用している。

$$L_s = K \cdot P_s \cdot (re - rc) \cdot A \cdot Imp / 3.6 \quad \text{..... (式 2-2-13)}$$

$$\frac{dP_s}{dt} = a - \frac{1}{3.6} K \cdot P_s (re - rc) \quad \text{..... (式 2-2-14)}$$

$$L_p = C \cdot P_p^m \cdot Q^n \cdot (Q - Q_c) \quad \text{..... (式 2-2-15)}$$

$$\frac{dP_p}{dt} = D - L_p \quad \text{..... (式 2-2-16)}$$

$$L_t = L_s + L_{pi} \quad \text{..... (式 2-2-17)}$$

ここで、	L_s	: 地表面流出負荷量	(g/s)
	K	: 地表面残存負荷流出係数	(1/mm)
	P_s	: 地表面残存負荷量	(kg/ha)
	re	: 有効降雨強度	(mm/hr)
	rc	: 限界降雨強度	(mm/hr)
	Imp	: 不浸透面積率	
	A	: 排水区流域面積	(ha)
	a	: 補給係数	(g/s/ha)
	L_p	: 管内流出負荷量	(g/s)
	C	: 負荷流出係数	
	P_p	: 管内堆積負荷量	(g)
	Q	: 流量	(m ³ /s)
	Q_c	: 限界流量	(m ³ /s)
	D	: 供給汚濁負荷量	(g/s)
	m, n	: 水質項目の負荷流出パターンにより異なる	
	L_t	: 総流出負荷量	(g/s)

式 2-2-15 における m 、 n は各水質項目の負荷流出パターンにより決定される指数であり、水質 5 項目における値は以下のとおりである。

表 2-2-1 各水質項目における m 、 n の値

項目	BOD	COD	SS	T-P	T-N
m	2	2	1	2	2
n	0	0	1	0	0

各パラメータの持つ意味およびその値の決定方法については、“合流式下水道改善策に関する調査 7” に詳細が解説されており、その適用例は 4 章で紹介する。

② 計算方法

以下に BOD のシミュレーションの場合を例にとり、基礎式の展開を示す。

晴天時の場合は、路面からの供給はないため、式 2-2-18 から、式 2-2-19 のように差分式に変形することができる。

BOD の場合、定数は $m=2$, $n=0$ であり、式 2-2-21 を式 2-2-20 に代入し整理すると、式 2-2-22 に示す未知数 Pp_2 に関する二次方程式となる。これを解くことにより、 Pp_2 が求まり、式 2-2-21 で流出負荷量 Li が算定されることとなる。これを計算ステップ数繰り返すことにより、順次流出負荷量が算定される。

$$\frac{dPp}{dt} = D - Li \quad \dots\dots\dots \text{(式 2-2-18)}$$

$$\frac{Pp_2 - Pp_1}{dt} = \frac{D_2 + D_1}{2} - \frac{Li_2 + Li_1}{2} \quad \dots\dots\dots \text{(式 2-2-19)}$$

$$Yp \equiv \frac{Li_2}{2} + \frac{Pp_2}{dt} + \frac{Li_1}{2} - \frac{Pp_1}{dt} - \frac{D_2 + D_1}{2} \quad \dots\dots\dots \text{(式 2-2-20)}$$

$$Li = C \cdot Pp^2 \cdot (Q - Qc) \quad \dots\dots\dots \text{(式 2-2-21)}$$

$$Yp \equiv \frac{C1(Q2 - Qc)}{2} Pp_2^2 + \frac{1}{dt} Pp_2 + \left(\frac{Li_1}{2} - \frac{Pp_1}{dt} - \frac{D_2 + D_1}{2} \right) \dots \text{(式 2-2-22)}$$

また、雨天時は、地表面からの供給を見込むため、式 2-2-13 の Ls の項が加わることになる。

Ls は、式 2-2-14 を式 2-2-23 の差分式にして整理した式 2-2-24 (未知数 Ps_2 の一次方程式) から Ps_2 を求め、式 2-2-25 より算定することができる。これを計算ステップ数繰り返すことにより、順次路面からの流出負荷量が算定される。 Li は既知の値として、式 2-2-16 に加えることとなる。

$$\frac{Ps_2 - Ps_1}{dt} = a - \frac{1}{3.6} K \cdot (Ps_2 - Ps_1)(re - rc) \quad \dots\dots\dots \text{(式 2-2-23)}$$

$$Ys \equiv \left(\frac{1}{dt} + \frac{K(re - rc)}{3.6} \right) Ps_2 - \left(\frac{Ps_1}{dt} + a + \frac{1}{3.6} K \cdot Ps_1(re - rc) \right) \dots\dots\dots \text{(式 2-2-24)}$$

$$Ls = K \cdot Ps \cdot (re - rc) \cdot A \cdot Imp / 3.6 \quad \dots\dots\dots \text{(式 2-2-25)}$$

(3) 管内水質モデル

管路モデルは計算が容易であること、また個別の管路における水質分布が流域全体の流出負荷量に及ぼす影響は無視できる程度であると考えられることから、従来の土研モデルに準じ、各タイムステップにおける総流出負荷量を流量で除して管路内の水質を求めるものとする。

すなわち、ある時間において、ネットワーク内の管路 n からの流出水量 Q_n に対し、汚濁負荷 $L_n(=L_{sn}+L_{pn})$ が流出する時、水質は

$$C_n(\text{mg/L}) = L_n(\text{g/s}) \div Q_n(\text{m}^3/\text{s})$$

で表現される。

ここで、 L_{sn} 、 L_{pn} はそれぞれ、管路 n の上流域からの流入も含めた地表面流出負荷量および管内流出負荷量を意味する。

3. モデル解析作業

3. モデル解析作業

本章では、下水研モデルを使った JAMSOOP ver. 1.1.0 における解析の考え方を示す。

3-1. 解析のプロセス

下水研モデルによる汚濁負荷流出解析を行うにあたっては、別途流量解析を行い、その結果を入力データとする。また、流域面積や負荷量、地表面パラメータ等の解析パラメータの設定を行う必要がある。

下水研モデルを用いた解析プロセスは、図 3-1-1 に示すとおりとなる。

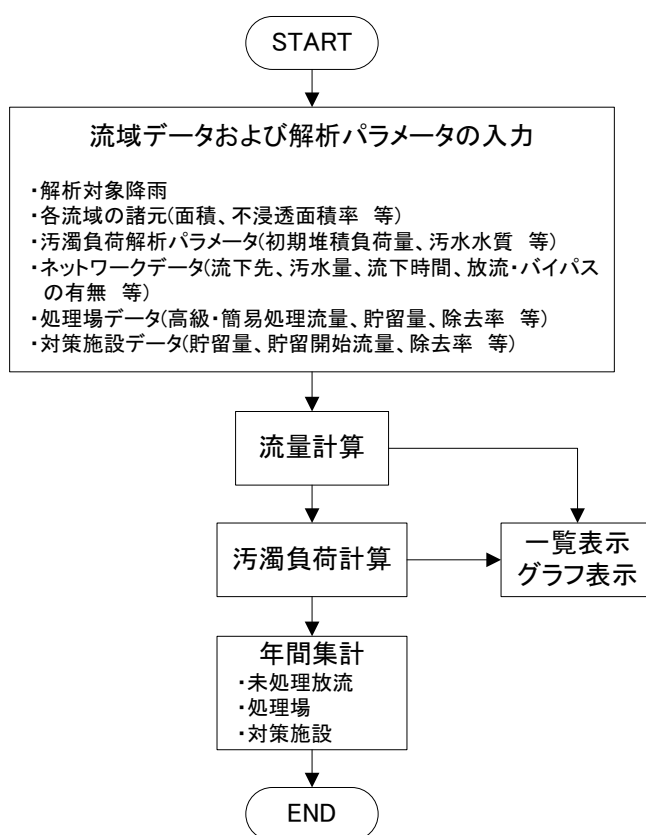


図 3-1-1 JAMSOOP ver.1.1.0 における解析のフロー図

今回、下水研モデルのプログラム本体と共に、下水研モデルによる汚濁負荷解析用のインターフェースソフト「JAMSOOP (Japanese Analysis Model for Sewer Overflow with Open Program)」の改良を行った。これを使用することにより、データの入力および解析結果の表示を簡便に行うことができる。

<集中型モデルと分布型モデルの計算手順の違い>

分布型汚濁負荷解析モデルの計算に必要となるパラメータの種類は、従来の集中型汚濁負荷計算とほぼ同じであるが、分布型汚濁負荷解析モデルの場合、モデル化する管路の分だけパラメータ数が多くなる。

このため、最初のモデルを作成するまでは集中型モデルよりも多大な作業が必要となるが、一旦、分布型モデルを構築してしまうと、そのモデルの変更は容易となる。

例えば、従来の集中型汚濁負荷モデルの場合、計算に用いる「不浸透面積率」、「地表面汚濁負荷パラメータ (Ps, K, rc)」、「晴天時水質」等は、対象流域の平均的な値を用いるため、局所的な特殊性を表現することができず、部分的に地域特性が変更となった場合、上記のパラメータをすべて設定し直す必要がある。これは対策施設の位置が変更になった場合も同様である。

一方、分布型汚濁負荷モデルの場合、浸透等の部分的に行う面的対策への対応、土地開発等により用途地域等の変更、工場等の特殊排水等は、その変更が必要な流域のみのパラメータ変更で対応が可能となり、このような地域特性を表現することができる*。

集中型モデル→平均値等で表現のため、再設定が必要

分布型モデル→局所的な対応が可能

例) 浸透による流出量の変更	→	不浸透面積率等
開発行為による用途地域の変更	→	Ps, K, rc
工場等による特殊排水	→	晴天時水質の変更

また、複数の同時施設対策を評価する場合、分布型モデルでは同一計算で自動的に施設評価を連動させることが可能であるが、集中型モデルでは個別の施設を別々の計算で行い、それらを連動させて評価する必要がある。

上記のように、分布型汚濁負荷モデルはモデル構築に多大な作業を要するが、様々なケースを想定し検討を行う場合、集中型モデルでは多大な作業量を要し、対応が困難となる。

※ 下水研モデルの基本的原理として流域別のパラメータ変更が可能であるが、今回作成した下水研モデルプログラムでは、簡便化のため流域別のパラメータ設定には対応しておらず、今後のプログラム改良に期待するものである(詳細は3-5を参照)。

3-2. 入力項目（境界条件）

下水研モデルの計算には様々なパラメータや境界条件、入力データが必要となる。

以下に、JAMSOOP ver.1.1.0における各項目の概要と、プログラム入力操作について示す。

- (1) 解析設定・降雨データ
- (2) 地表面流出解析パラメータ
- (3) 等到達時間パラメータ
- (4) ネットワークデータ
- (5) 汚濁負荷解析パラメータ
- (6) 処理場パラメータ
- (7) 対策施設パラメータ

下水研モデルでは、境界条件として表 3-2-1 に示す項目が必要となる。

表 3-2-1 必要入力データ

データ項目	データ内容
① 解析設定・降雨データ	年間降雨データ 解析ステップ
② 地表面流出解析パラメータ	流域面積 不浸透面積率 浸透域浸透能 浸透域くぼ地割合 流達時間 等
③ 等到達時間パラメータ	等到達時間割合(10%間隔)
④ ネットワークデータ	流下先ブロック番号 雨水吐き・バイパス等の有無 遮集倍率 バイパス先ブロック番号 晴天時平均汚水量 等
⑤ 汚濁負荷解析パラメータ	晴天時平均水質 限界流量 地表面残存負荷流出係数 初期地表面残存負荷量 等
⑥ 処理場パラメータ	高級処理量 簡易処理量 貯留容量 処理率 等
⑦ 対策施設パラメータ	貯留開始流量 貯留容量 遮集量 処理率等

各項目のデータ構成とデータファイルの書式、「JAMSOOP ver.1.1.0」における入力例について、以下に示す。

なお、「JAMSOOP ver.1.1.0」の操作に関する詳細は「JAMSOOP ver.1.1.0 操作説明書」¹⁰⁾を参照されたい。

(1) 解析設定・降雨データ

計算ステップ数、解析対象降雨の日時および降雨データを入力する。

	A	B	C	D	E	F	G	H	I	J	K	L
1	分/1ステップ	10										
2	年	月	日	時	分	降雨データ						
3	2000	1	1	0	0	0	0	0	0	0	0.5	0
4	2000	2	1	0	0	0	0	0	0	0.5	0	0
5	2000	3	1	0	0	0	0	0	0.5	0	2	0.5
6	2000	4	1	0	0	0	0	0	0	0.5	0.5	0.5
7	2000	5	1	0	0	0	0	0	0.5	1.5	1.5	0
8	2000	6	1	0	0	0	0	0	0.5	0	0.5	0
9	2000	7	1	0	0	0	0.5	0	0	0.5	0	0
10	2000	8	1	0	0	0.5	0	0.5	0.5	0	0.5	0
11	2000	9	1	0	0	0	0	0	0.5	0.5	0.5	0
12	2000	10	1	0	0	0	0	0	0	0.5	0	0

データ入力内容：

1行目 : 分/1ステップ (1ステップあたりの時間を入力)

2行目 : 年 月 日 時 分 降雨データ (3行目以降のデータ項目の表示)

3行目以降 : 2000 1 1 0 0 0 0 0 0 0 0.5 0 …

降雨開始年、月、日、時、分、降雨時系列データ

図 3-2-1 エクセルでの入力画面(CSVファイル)

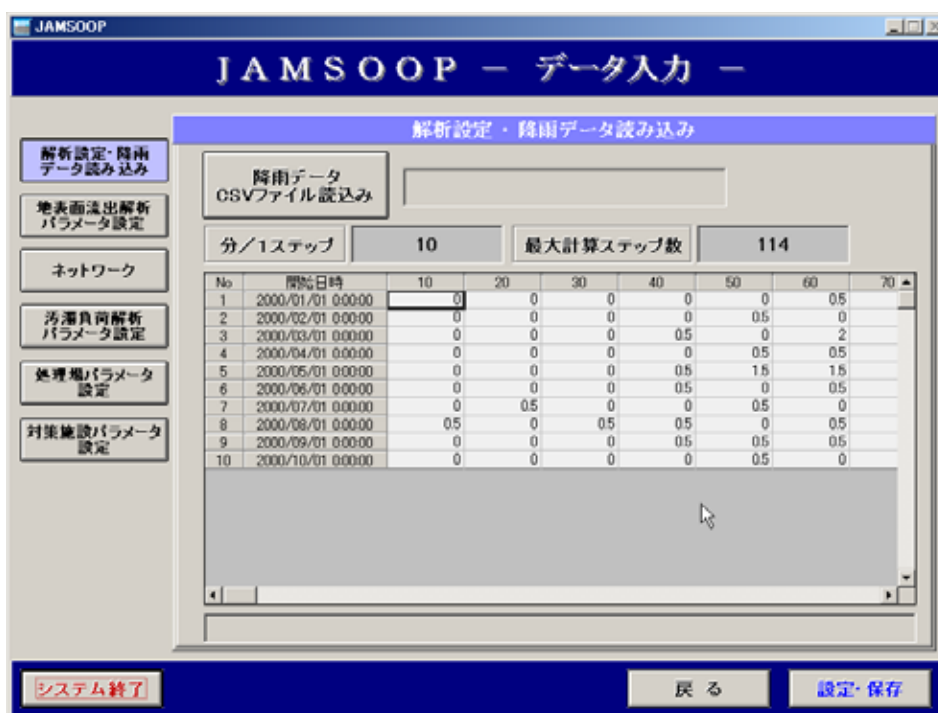


図 3-2-2 JAMSOOP ver.1.1.0 での入力画面

それぞれの入力項目は以下のパラメータを表す。

- ・ 分/1ステップ … モデル計算の1ステップの時間間隔
- ・ 最大計算ステップ数 … 複数降雨中で最も長い降雨のステップ数

(2) 地表面流出解析パラメータ

地表面流出解析パラメータは、以下のようなシートを作成し、各流域の諸元を入力する。

	A	B	C	D	E	F	G	H	I	
1		流域面積	不浸透面積	浸透域浸透	浸透域凹地	浸透域凹地	不浸透域	不浸透域	到達時間	
2	単位	ha	-	mm/hr	%	mm	%	mm	分	
3	1	1	0.5	10	80	6	60	2	10	
4	2	2	0.5	10	80	6	60	2	10	
5	3	1	0.5	10	80	6	60	2	10	
6	4	2	0.5	10	80	6	60	2	10	
7	5	1	0.5	10	80	6	60	2	10	
8	6	2	0.5	10	80	6	60	2	10	
9	7	1	0.5	10	80	6	60	2	10	
10	8	2	0.5	10	80	6	60	2	10	
11	9	1	0.5	10	80	6	60	2	10	
12	10	0	0	0	0	0	0	0	10	

データ入力内容：
 1行目：データ項目
 2行目：単位
 3行目以降：各流域の諸元
 1 1 0.5 10 80 6 60 2 10
 流域ブロック番号、流域面積、不浸透面積率、浸透域浸透能、浸透域くぼ地割合、浸透域くぼ地貯留高、
 不浸透域くぼ地割合、不浸透域くぼ地貯留高、到達時間

図 3-2-3 エクセルでの入力画面(CSV ファイル)

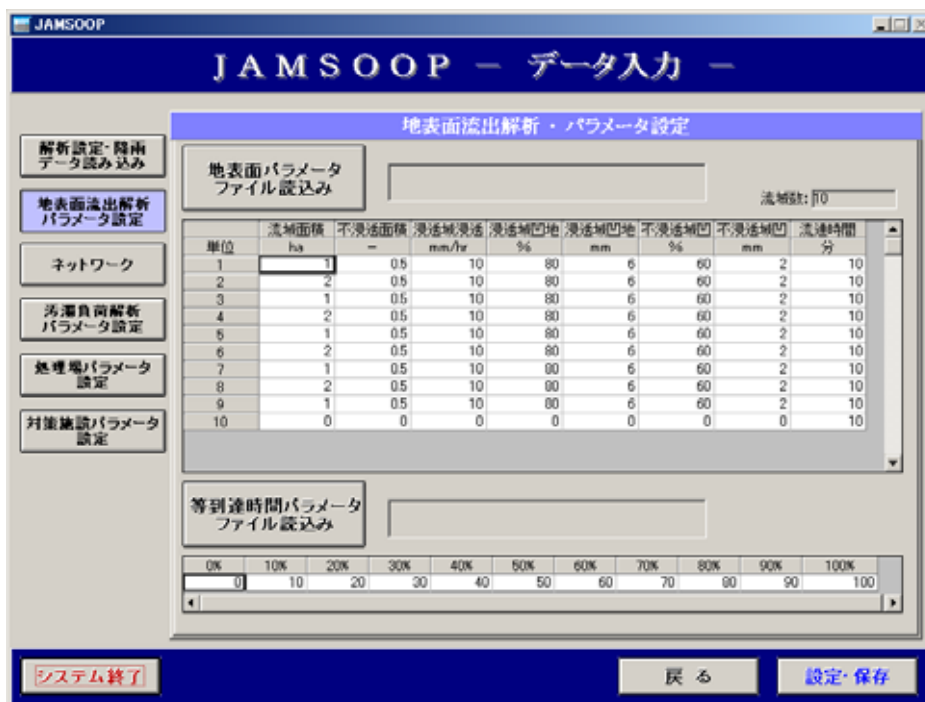


図 3-2-4 JAMSOOP ver.1.1.0 での入力画面

(3) 等到達時間パラメータ

等到達時間パラメータは、以下のようなシートを作成し、各等到達時間区分の面積割合を入力する。

等到達時間面積割合は、例えば流達時間が 30 分である流域を対象とした場合、降った雨が流末まで到達する面積割合を到達時間 0～3 分(10%)、3～6 分(20%)、…27～30 分(100%)で区切って表したものである。

	A	B	C	D	E	F	G	H	I	J	K
1	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
2	0	10	20	30	40	50	60	70	80	90	100
3											

データ入力内容：
 1 行目：等到達時間区分
 2 行目：等到達時間面積率(%)

図 3-2-5 エクセルでの入力画面(CSV ファイル)

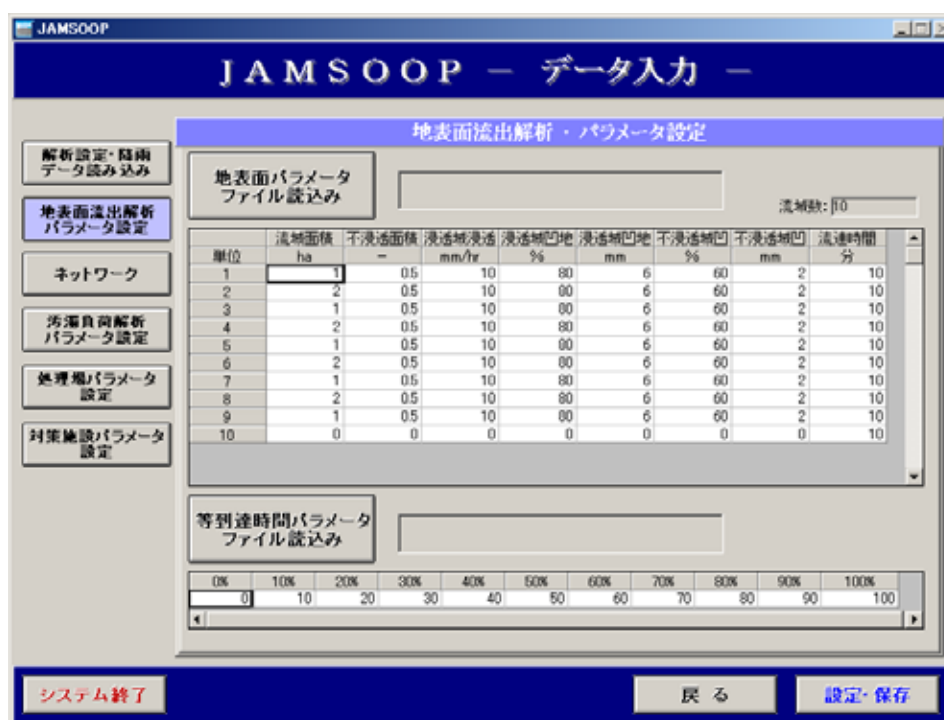


図 3-2-6 JAMSOOP ver.1.1.0 での入力画面

(4) ネットワークデータ

管路網のデータは以下のようなシートを作成し、流下先ブロックや雨水吐きの有無等を入力する。

ただし、遮集倍率、バイパス先ブロック番号、晴天時最大汚水量はそれぞれ雨水吐きやバイパスのあるブロックのみ設定する。

1	A	B	C	D	E	F	G	H	I	J
2	単位	流下先 ブロック番号	雨水吐きなど(0:なし)	遮集倍率 Q	バイパス先 ブロック番号	晴天時平均汚水量 m3/s	晴天時最大汚水量 m3/s	下流への流下時間 分	SQカーブ式:K	SQカーブ式:P
3	1	2	0	0	0	0.0001	0	0	0.08	0.7
4	2	4	0	0	0	0.0002	0	0	0.08	0.7
5	3	4	0	0	0	0.0001	0	0	0.08	0.7
6	4	8	1	3	0	0.0002	0.0012	0	0.08	0.7
7	5	6	0	0	0	0.0001	0	0	0.08	0.7
8	6	7	3	3	9	0.0002	0.0006	20	0.08	0.7
9	7	8	0	0	0	0.0001	0	0	0.08	0.7
10	8	9	0	0	0	0.0002	0	0	0.08	0.7
11	9	10	0	0	0	0.0001	0	0	0.08	0.7
12	10	0	0	0	0	0	0	0	0.08	0.7
13										

データ入力内容：

1 行目：データ項目

2 行目：単位

3 行目以降：各流域の諸元

1 2 0 0 0 0.001 0 0.08 0.7

流域ブロック番号、流下先ブロック番号、雨水吐きなどの有無(0：なし、1：雨水吐き、2：ポンプ場、3：バイパス)、バイパス先ブロック番号、晴天時平均汚水量、晴天時最大汚水量、下流への流下時間、SQカーブ式定数 K、SQカーブ式定数 p

図 3-2-7 エクセルでの入力画面 (CSV ファイル)

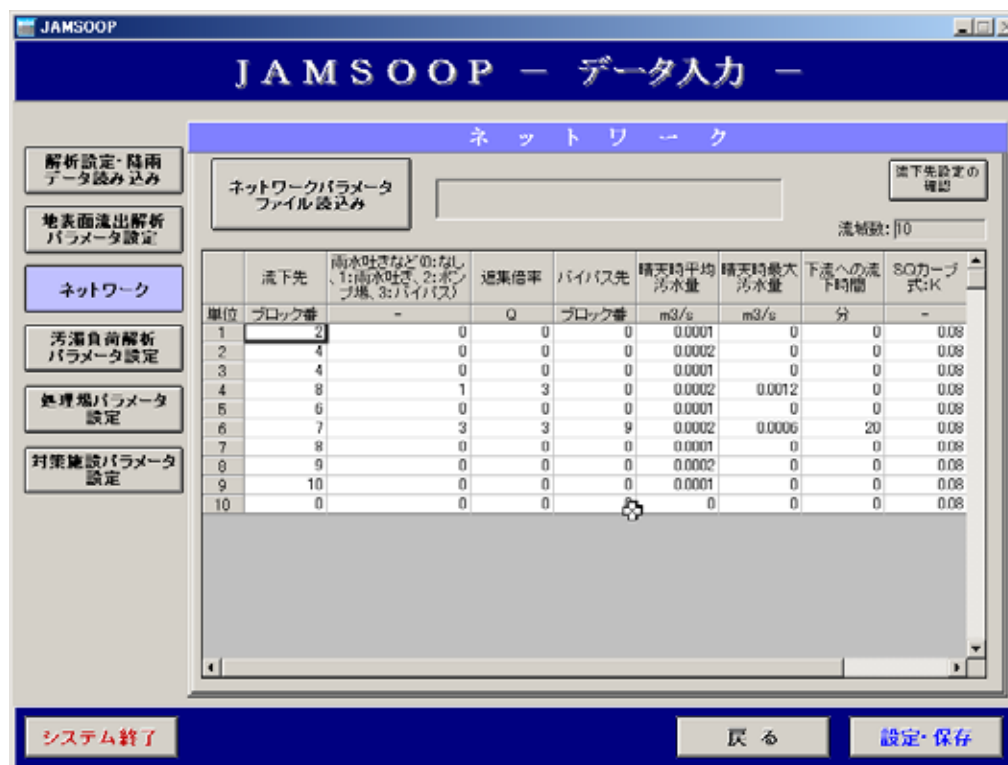


図 3-2-8 JAMSOOP ver.1.1.0 での入力画面

(5) 汚濁負荷解析パラメータ

管内汚濁負荷パラメータおよび地表面汚濁負荷パラメータについて、以下のようなシートを作成し入力する。JAMSOOPでは、水質5項目を同時に入力することが可能である。

	A	B	C	D	E	F	G
1			BOD	COD	SS	T-P	T-N
2	晴天時平均水質	mg/L	200	150	180	5	30
3	限界流量	m ³ /s/ha	0.0001	0.0001	0.0001	0.0001	0.0001
4	地表面残存負荷流出係数	l/mm	0.1	0.15	0.4	0.35	0.01
5	初期地表面残存負荷量	kg/ha	5	4	5	0.35	2.9
6	限界降雨強度	mm/hr	2	2	2	2	2
7	補給係数	g/s/ha	0.029	0.023	0.029	0.002	0.017
8							

データ入力内容：
 1行目：データ項目（水質5項目）
 2行目：晴天時平均水質(mg/L)
 3行目：限界流量(m³/s/ha)
 4行目：地表面残存負荷流出係数(l/mm)
 5行目：初期地表面残存負荷量(kg/ha)
 6行目：限界降雨強度(mm/hr)
 7行目：補給係数(g/s/ha)

図 3-2-9 エクセルでの入力画面 (CSV ファイル)

The screenshot shows the JAMSOOP software interface for data input. The window title is "JAMSOOP - データ入力". The main area is titled "汚濁負荷解析・パラメータ設定率". On the left, there are several menu items: "解析設定・降雨データ読み込み", "地表面流出解析パラメータ設定", "ネットワーク", "汚濁負荷解析パラメータ設定" (highlighted), "処理場パラメータ設定", and "対策施設パラメータ設定". The main data entry area contains a table with the following data:

		BOD	COD	SS	T-P	T-N
晴天時平均水質	mg/L	200	150	180	5	30
限界流量	m ³ /s/ha	0.0001	0.0001	0.0001	0.0001	0.0001
地表面残存負荷流出係数	l/mm	0.1	0.15	0.4	0.35	0.01
初期地表面残存負荷量	kg/ha	5	4	5	0.35	2.9
限界降雨強度	mm/hr	2	2	2	2	2
補給係数	g/s/ha	0.029	0.023	0.029	0.002	0.017

Below the table, there are two buttons: "管内パラメータ" (In-pipe parameters) and "地表面パラメータ" (Surface parameters). At the bottom, there is a "計算項目" (Calculation Items) table:

計算項目	BOD	COD	SS	T-P	T-N
m	2	2	1	2	2
n	0	0	1	0	0

At the bottom of the window, there are buttons for "システム終了" (System End), "戻る" (Back), and "設定・保存" (Settings/Save).

図 3-2-10 JAMSOOP ver.1.1.0 での入力画面

それぞれの入力項目は以下のパラメータを表す。

(管内汚濁負荷パラメータ)

- ・ 晴天時日平均水質 …… 晴天時汚水の平均水質(観測値を入力)(mg/L)
- ・ 限界流量 …… 管内汚濁負荷が流出し始める最小の流量(m³/s/ha)

(地表面汚濁負荷パラメータ)

- ・ 地表面残存負荷流出係数 : K …… 降雨量あたりで汚濁負荷量が流出する割合
(1/mm)
- ・ 初期地表面残存負荷量 : Ps0 …… 流域に堆積している汚濁負荷量(kg/ha)
- ・ 限界降雨強度 : rc …… 地表面汚濁負荷が流出し始める最小の降雨強度(mm/hr)
- ・ 補給係数 : a …… 無降雨時に汚濁負荷の堆積が回復する割合(kg/s/ha)

各パラメータの設定方法については、4章で解説する。

なお、JAMSOOP ver.1.1.0 の入力画面に示すように、各水質項目における汚濁負荷流出式内の係数 m、n(表 2-2-1 を参照)は、固定値として与えられる。

(6) 処理場パラメータ

処理場の設定について、以下のようなシートを作成し入力する。一つの対象流域について複数の処理場の設定も可能である。

また、処理場パラメータを設定しない場合でも解析は可能である。

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	処理場名	場所	高級処理水量	簡易処理水量	貯留容量	高級処理率BOD	簡易処理率BOD	高級処理率COD	簡易処理率COD	高級処理率SS	簡易処理率SS	高級処理率T-P	簡易処理率T-P	高級処理率T-N	簡易処理率T-N
2	処理場名	流域番号	m ³ /s	m ³ /s	m ³	%	%	%	%	%	%	%	%	%	%
3	サンプル処理場1	10	0.26	0.52	1000	90	30	80	30	90	60	60	20	60	20

データ入力内容：
 1行目：パラメータ項目
 2行目：単位
 3行目以降：処理場名、流域番号、高級処理水量、簡易処理水量、貯留容量、高級処理率、簡易処理率
 (BOD、COD、SS、T-N、T-P)

図 3-2-11 エクセルでの入力画面(CSV ファイル)

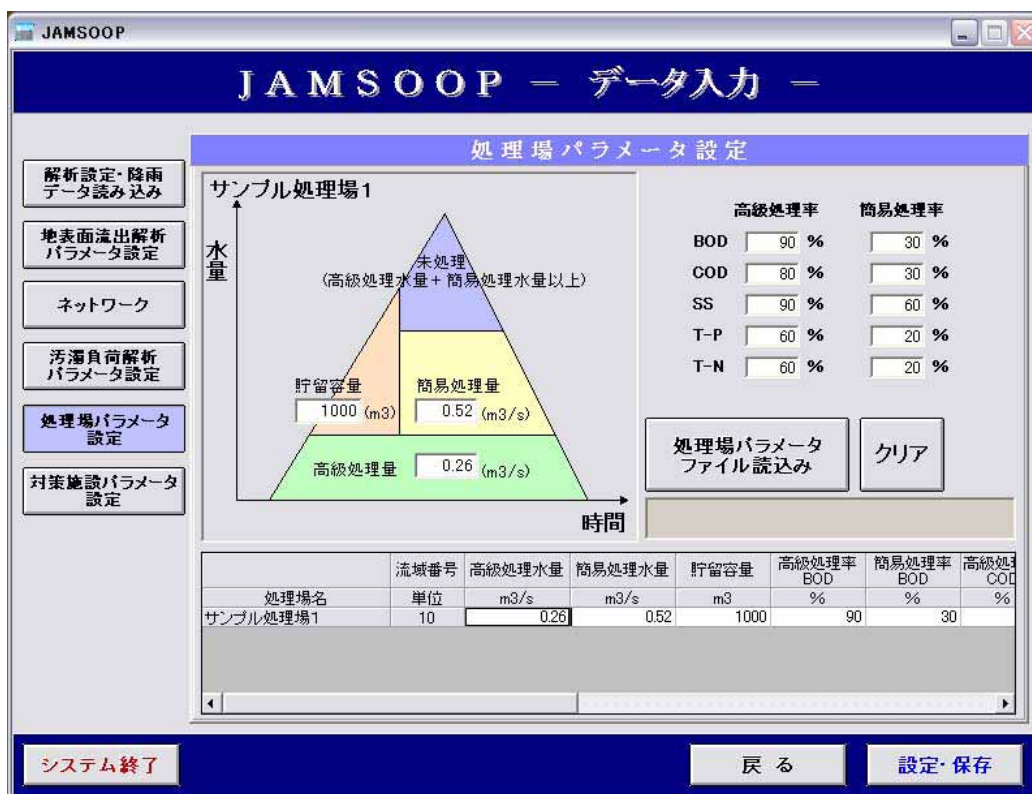


図 3-2-12 JAMSOOP ver.1.1.0 での入力画面

(7) 対策施設パラメータ

対策施設(貯留池)の設定について、以下のようなシートを作成し入力する。一つの対象流域について複数の貯留池の設定も可能である。

また、対策施設パラメータを設定しない場合でも解析は可能である。

	A	B	C	D	E	F	G	H	I	J	K
1	対策施設名	場所	貯留開始流量	貯留上限流量	貯留容量	3Q	施設処理率BOD	施設処理率COD	施設処理率SS	施設処理率T-P	施設処理率T-N
2	施設名	流域番号	m3/s	m3/s	m3	m3/s	%	%	%	%	%
3	対策施設1	4	0.12		1000	0.36	90	80	90	60	60

データ入力内容：

1行目：パラメータ項目

2行目：単位

3行目以降：対策施設名、流域番号、貯留開始流量、貯留上限流量(設定なしも可)、貯留容量、3Q、施設処理率(BOD、COD、SS、T-N、T-P)

図 3-2-13 エクセルでの入力画面(CSV ファイル)

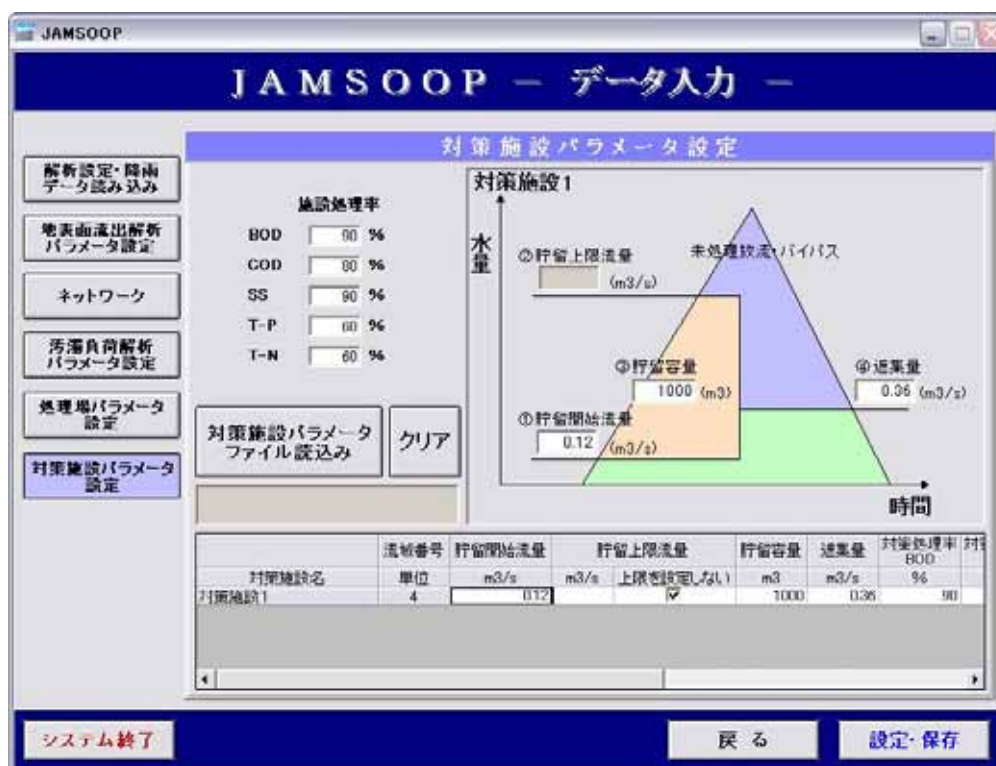


図 3-2-14 JAMSOOP ver.1.1.0 での入力画面

3-3. 計算手順

本解析の手順は、

分布型流量計算 → 分布型汚濁負荷計算 → 結果の出力 となる。

JAMSOOP ver.1.1.0により、データの入力、計算の実行、結果の出力をインターフェース上で行うことが可能である。

「JAMSOOP ver.1.1.0」の設定画面で各データを入力後、「流量計算実行」ボタンおよび「汚濁負荷計算実行」ボタンを押すと、計算・集計が一括で行われる。

ただし、流量計算を実行した後でなければ汚濁負荷計算は実行できない。



図 3-3-1 JAMSOOP ver.1.1.0 における計算の実行

3-4. 出力項目

解析結果は、解析終了後に集計シートおよび簡易グラフに出力される。また、結果一覧表をエクセルへ貼り付け、編集することが可能である。

解析結果については、流量および5水質項目について、各流域・各降雨の時系列データ、処理場の高級処理、簡易処理、貯留量等の各項目の時系列データ、対策施設の各項目の時系列データの一覧表示、グラフ表示をすることができる。

また、解析対象とした全降雨における未処理放流、処理場、対策施設の一括集計を行うことができる。

	10	20	30	40	50	60	70	80	90	100	110	120	130	140
1	0.000100	0.000100	0.000100	0.000854	0.003839	0.010461	0.004976	0.000100	0.001594	0.002267	0.000100	0.002170	0.002186	0.002
2	0.000300	0.000300	0.000300	0.001849	0.006265	0.024014	0.018274	0.004923	0.007327	0.004776	0.001830	0.006656	0.005387	0.006
3	0.000100	0.000100	0.000100	0.000854	0.003839	0.010461	0.004976	0.000100	0.001594	0.002267	0.000100	0.002170	0.002186	0.002
4	0.000600	0.000600	0.000600	0.001200	0.001200	0.001200	0.001200	0.001200	0.001200	0.001200	0.001200	0.001200	0.001200	0.001
5	0.000100	0.000100	0.000100	0.000854	0.003839	0.010461	0.004976	0.000100	0.001594	0.002267	0.000100	0.002170	0.002186	0.002
6	0.000300	0.000300	0.000300	0.001800	0.001800	0.001800	0.001800	0.001800	0.001800	0.001800	0.001800	0.001800	0.001800	0.001
7	0.000400	0.000400	0.000400	0.001230	0.005139	0.012506	0.006530	0.000900	0.004523	0.003049	0.001530	0.004471	0.003960	0.004
8	0.001200	0.001200	0.001200	0.002940	0.009743	0.026442	0.021313	0.007621	0.009722	0.007903	0.005093	0.008993	0.009462	0.010
9	0.001300	0.001300	0.001300	0.002240	0.006319	0.018680	0.027170	0.022471	0.018725	0.028273	0.028248	0.013965	0.013965	0.015
10	0.001300	0.001300	0.001300	0.001300	0.001650	0.003890	0.012548	0.024309	0.024912	0.019889	0.024084	0.029052	0.019707	0.012

図 3-4-1 解析結果の一覧表示

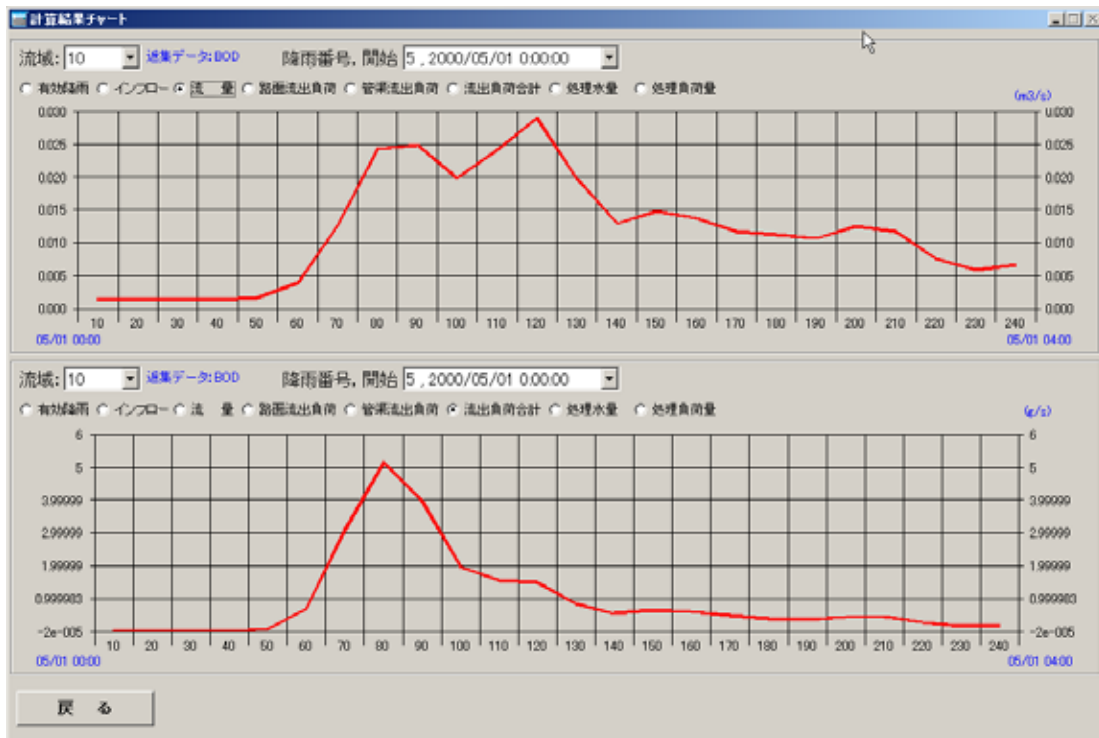


図 3-4-2 解析結果のグラフ表示

計算結果

降雨番号, 開始日時: 5, 2000/05/01 0.00:00

シミュレーション計算結果一覧表

有効降雨(mm/10分) | インフロー(m³/s) | 流量(m³/s) | 路面流出負荷(q/s) | 管渠流出負荷(q/s) | 流出負荷合計(q/s) | 処理場・対策施設 | 年間集計結果

年間集計結果(年間合計)

表示単位を切り替える: 千m³・トン | m³・kg

処理場計算結果		高効率処理水量	簡易処理水量	貯留水量	直接排水量	高効率処理流入負荷量	簡易処理流入負荷量	高効率処理流出負荷量	簡易処理流出負荷量	貯留流入負荷量	貯留流出負荷量(高効率処理)	直接排水負荷量
処理場名	単位	(千m³)	(千m³)	(千m³)	(千m³)	(トン)	(トン)	(トン)	(トン)	(トン)	(トン)	(トン)
サンプル処理場1	10	0.5	0.0	1.9	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0
対策施設計算結果		貯留水量	直接排水量	貯留流入負荷量	貯留流出負荷量(高効率処理)	直接排水負荷量	排水回数					
対策施設名	単位	(千m³)	(千m³)	(トン)	(トン)	(トン)						
対策施設1	4	1.7	0.0	0.1	0.0	0.0	0					
雨水吐き計算結果		直接排水量	直接排水負荷量	排水回数								
雨水吐き	単位	(千m³)	(トン)									
雨水吐き	4	0.0	0.0	0								
ポンプ場計算結果		直接排水量	直接排水負荷量	排水回数								
ポンプ場	単位	(千m³)	(トン)									
ポンプ場												

戻る | 条件入力画面へ

※Ctrl+C でコピー出来ます。 | Chart

図 3-4-3 年間集計結果の表示

3-5. プログラムの制限事項

下水研モデルの基本的原理は、個別の流域における汚濁負荷流出を積上げる分布型であり、汚濁負荷の堆積や流出に関して各流域で異なるパラメータ値の設定、負荷流出計算が可能である。

しかし、本検討で下水研モデルをプログラム化するにあたりモデル開発を優先したことや、現時点での解析上の需要を考慮し、いくつかの項目で簡便化を図っている。解析においてより詳細な設定を行いたい場合、これらの項目についてプログラムや入力ファイルの修正が必要となる。

これらの制限事項は、基本的にはプログラムを改良することで対応可能であり、将来的な改良、発展に期待するものである。

(1) 管路および地表面パラメータ設定における制限事項

以下に示す管路および地表面パラメータは、検討対象流域の特性により異なるものであり、現地での水質調査結果や土地利用別に定められた値の積上げにより決定されるものである。

表 3-5-1 パラメータ一覧

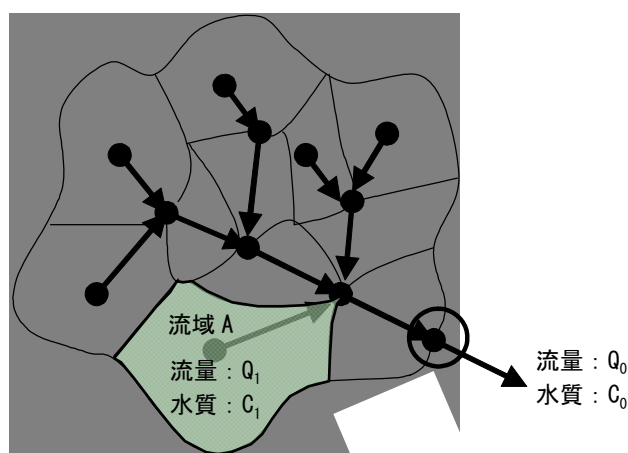
項目	パラメータ	記号	パラメータの意味	設定方法の概要
管路パラメータ	初期管内堆積負荷量	Pp0	管内堆積負荷量の初期値	晴天時日発生負荷量の 1/2 晴天時水質から設定
	負荷流出係数	C	降雨強度および残存負荷量にかかる係数 管路からの単位時間当たりの流出負荷量を決定	$C=1.2 \times DL/Q/P_{p0}^2$ で算出 (SS 以外) $C=DL/Q^2/P_{p0}$ で算出 (SS)
地表面パラメータ	初期地表面残存負荷量	Ps0	地表面堆積負荷量の初期値	雨天時分流雨水の水質調査結果や土地利用別の既往の調査結果に基づき設定
	地表面残存負荷流出係数	K	降雨強度および残存負荷量にかかる係数 地表面からの単位時間当たりの流出負荷量を決定	土地利用別の既往の調査結果から設定
	補給係数	a	堆積負荷量の回復速度	1~2 日で初期の堆積状態に戻ると仮定 初期地表面残存負荷量/1~2 日間で設定

検討対象流域の中で局所的に特別な土地利用があったり、他流域と水質が大きく異なる排水が行われていたりする流域については、個別の水質調査や土地利用面積調査等により、これらのパラメータを他の流域と区別して設定するのが望ましい。

しかし、今回開発したプログラムでは個別の流域における管路および地表面パラメータの設定には対応しておらず、全流域で一律のパラメータ設定を用いることとしている。

これを修正するには、地表面パラメータであれば図 3-2-9 に示す CSV ファイルや図 3-2-10 に示す JAMSOOP 上のウィンドーで入力を行う現況設定について、各流域のパラメータ値(土地利用面積率などから算出可能)を羅列した CSV ファイルを作成し、これを読み込むように変更するという方法が考えられる。

また、管路パラメータの場合、複数点の水質調査結果をモデルに組み込むには負荷量の収支を考慮する必要があるため(図 3-5-1)、晴天時水質と晴天時流量から負荷量を算出し、観測地点の上下流における汚濁負荷収支を考慮した上で各流域にパラメータ設定をするように、負荷収支計算過程をプログラムに加えて修正する必要がある。



流域 A での流量を Q_1 、水質を C_1 、最下流での流量を Q_0 、水質を C_0 とした場合、負荷量の収支を考慮すると、流域 A 以外の流域からの流量は $Q_0 - Q_1$ 、水質 $(C_0 Q_0 - C_1 Q_1) / (Q_0 - Q_1)$ となる。

図 3-5-1 晴天時水質の負荷収支の概念

4. パラメータの考え方(設定方法)

4. パラメータの考え方（設定方法）

下水研モデルのパラメータの設定方法は、基本的に従来の集中型モデル(土研モデル)におけるパラメータ設定方法に準ずるものとする。

なお今回開発した下水研モデルプログラム「JAMSOOP」においては、晴天時平均水質、日平均汚水量といった関連する入力値から、各流域におけるパラメータ値を自動的に算出する。

<暫定指針におけるパラメータ決定法>

- i) $Pp0$ を日発生負荷量の $1/2$ (晴天時平均水質 \times 日流量 $\times 1/2$)とする。
- ii) $C = \text{晴天時平均水質} \times 1.2 / Pp0^2$ とする。SS の場合には $C = \text{晴天時平均水質} / (Pp0 \cdot \text{日平均流量})$ とする。
- iii) 限界流量 qc は、最小流量程度とする。

「合流式下水道改善対策指針と解説—2002年版—, (社)下水道協会」¹⁾より

以下に下水研モデルで用いる各種パラメータの設定方法について、管路パラメータと地表面パラメータに分けて説明する。

4-1. 管路パラメータの設定方法

従来の集中型汚濁負荷解析においては、評価点より上流を代表するパラメータ 1 点を設定するのみであったが、下水研モデルにおけるパラメータは、計算する流域モデルの数だけ設定が必要となる*。

以下に、従来の集中型モデルにおけるパラメータ決定法に準じたパラメータの設定方法を示す。

- (1) 初期管内堆積負荷量(Pp0)
- (2) 負荷流出係数 (C)
- (3) 限界流量(qc)
- (4) 流量データ
- (5) 管路内水量 (Vol)
- (6) m、n

※ 今回開発した下水研モデルプログラムでは、個別の流域で異なる観測値を用いたパラメータ設定には対応していない(詳細は「3-5. プログラムの制限事項」を参照)。

(1) 初期管内堆積負荷量(Pp0)

Pp0 は流域毎に、各地点における日発生負荷量の 0.5 倍とする。

ある 1 地点における実測値等があり、1 地点の日発生負荷量が把握できている場合、既知点の単位面積当たり日発生負荷量に、各路線の集水面積を掛け合わせ、各路線の日発生負荷量を設定する。

なお JAMSOOP では、入力パラメータである日平均汚水量(各流域で固有の値)と晴天時平均水質(共通の値)から、初期管内堆積負荷量を自動的に計算する設定としている。

(2) 負荷流出係数 (C)

集中型シミュレーションと同様に、管路毎に雨天時最大水質 = $C^* \cdot Pp0^2$ となる C^* を求め、 C^* と Pp0 で晴天時シミュレーションを行い、最終的に安定した $Pp0^*$ を求め、 $C^* \cdot Pp0^2 = C \cdot Pp^{*2}$ となる C を決定する。SS の場合には $C^* \cdot (Pp0 \times \text{日平均流量}) = C \cdot (Pp^* \times \text{日平均流量})$ とする。

なお JAMSOOP では、入力パラメータである晴天時平均水質と (1) で求めた Pp0 により、下式を用いて簡易的に設定する。

$$C = \text{晴天時平均水質} \times 1.2 / Pp0^2 \quad \dots \text{(SS 以外の場合)}$$

$$C = \text{晴天時平均水質} / (Pp0 \cdot \text{日平均流量}) \quad \dots \text{(SS の場合)}$$

(3) 限界流量(q_c)

限界流量は雨天時の汚濁負荷流出が始まる最小の流量である。

JAMSOOP では面積当たりの限界流量を共通のパラメータとして入力し、これを自動的に流域面積に乗じて各流域の限界流量を設定する。

(4) 流量データ

下水研モデルでは、各路線の流量データは、別途、分布型雨水流出解析モデル等で計算した値を入力値とする必要がある。

ただし JAMSOOP においては、下水研モデルの事前計算として修正 RRL 法による流量解析が可能であり、これにより得られた流量データを用いる。

流量データとしては、流域モデルの計算に用いる流量と管路モデルの計算に用いる流量の2つの時系列の流量が必要となる。考え方としては、図 4-1-1 に示すとおりとなる。

- ・ 管路が上流端で自流域を持つ場合、流域モデルからの流量 (q_1) が管路の流量 ($Q_1=q_1$) となる。
- ・ 自流域を持つ管路へ接続する場合、接続先の自流域からの流量 (q_2) と流入する流量 (Q_1) の合算が、接続先の流量 ($Q_2=q_2+Q_1$) となる。
- ・ 管路が自流域を持たない場合、その管路の流域モデルにおける流量は“0”となる ($q_4=0$)。
- ・ 管路が自流域を持たない場合、その管路の管路モデルにおける流量は、流入管の流量の合算となる ($Q_4=Q_3+Q_2$)。

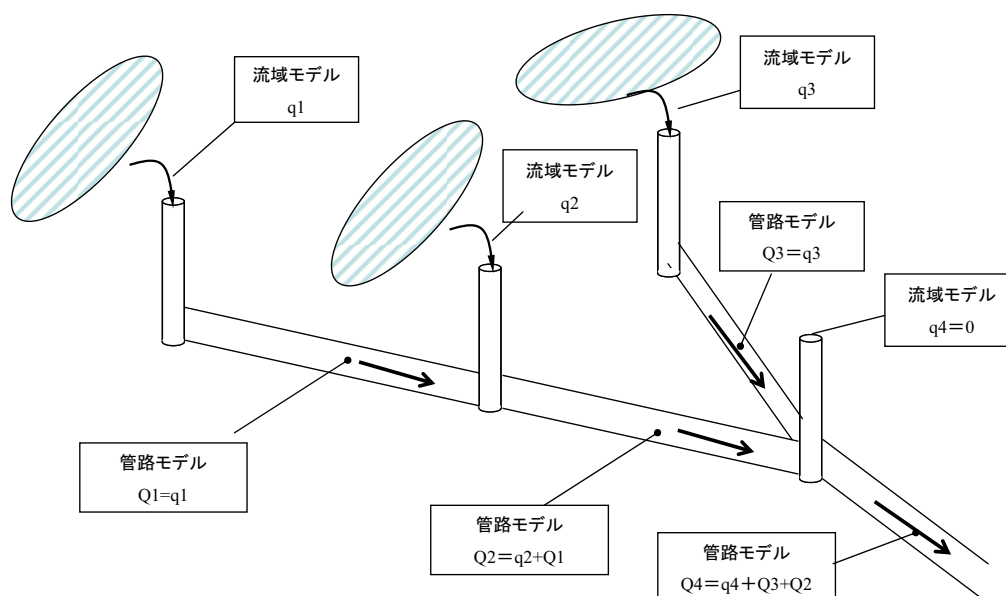


図 4-1-1 流量の考え方

なお、使用する流量データは、キャリブレーション等がなされ、十分に再現性のあるデータでなければならない。

(5) m、n

式 2-2-15(以下に再掲)の管内堆積負荷量 P_p および流量 Q に掛かる指数 m 、 n は各水質項目の負荷流出パターンにより決定される指数であり、各水質項目により固有の値である。

$$L_i = C \cdot P_p^m \cdot Q^n \cdot (Q - Q_c) \dots \dots \dots \text{(式 2-2-15、再掲)}$$

一般に雨天時における流出水量と汚濁負荷量の関係は、図 4-1-2 に示すように時計回りのループを描くといわれている。また、SS のループは、BOD や COD のそれと比較して直線的であり、流出特性が異なることが知られている。

時計回りのループを描くということは、雨天時流出水量が同じでも前半の流量による流出負荷量の方が後半の流量による流出負荷量よりも多いことを示す。

SS のループが BOD や COD のループと比べ直線的であるのは、地表面からの供給負荷量が相対的に多く、流出後半においても流出負荷量がある程度維持されているためと考えられている。

BOD 等の負荷量は流量の n 乗に比例するものとされ、既往の調査⁴⁾において、BOD、COD、SS の指数部 (m, n) は BOD、COD : $m=2$ 、 $n=0$ 、SS : $m=1$ 、 $n=1$ とされている。

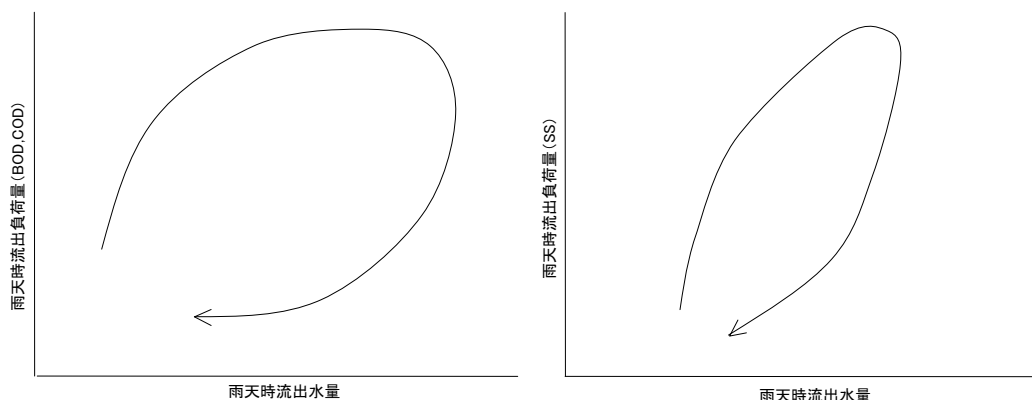


図 4-1-2 雨天時流出水量と流出負荷量の関係

窒素、リンの指数は、別途行った観測結果より BOD、COD と負荷流出パターンが類似していたため、 $m=2$ 、 $n=0$ とした。

表 4-1-1 各水質項目における m 、 n の値

項目	BOD	COD	SS	T-P	T-N
m	2	2	1	2	2
n	0	0	1	0	0

4-2. 地表面パラメータ設定に関する参考資料

解析に用いる地表面パラメータは対象流域に固有の値であり、一般的には雨天時水質調査等の結果から設定する。但し、既往の研究⁶⁾により、BOD、COD、SSについては土地利用ごとのパラメータ値の目安が示されている。

また、T-P、T-Nについては別途行われた分流雨水の水質調査結果が参考となる。

(1) BOD, COD, SS の場合

(2) T-P, T-N の場合

「分布型汚濁負荷モデルのパラメータの決定法は、従来の集中型モデルにおけるパラメータ決定法に準ずる方法を基本とする。」としたが、従来の集中型モデルにおいても、地表面パラメータの設定法は明記されていない。

以下に、地表面パラメータの設定例を示す。

(1) BOD, COD, SS の場合

BOD および COD, SS の地表面残存負荷流出係数(K)と初期地表面残存負荷量 (Ps0) は、土木研究所における既存の研究結果⁶⁾により、表 4-2-1, 2 に示すように値が示されている。

下水研モデルによるシミュレーションにおいては、このデータを基に集水エリアにおける用途地域の面積割合を用いて加重平均し、「地表面残存負荷流出係数(K)」、「初期地表面残存負荷量(Ps0)」を求めた。

表 4-2-1 地表面残存負荷流出係数(K) (単位：1/mm)

工種	用途	BOD	COD	SS
屋根		0.25~0.30 (0.28)	0.25~0.30 (0.28)	0.50~0.65 (0.58)
地表面	住宅地	0.10~0.12 (0.11)	0.09~0.13 (0.11)	0.13~0.16 (0.15)
地表面	商店街	0.10~0.20 (0.15)	0.10~0.12 (0.11)	0.10~0.20 (0.15)
地表面	工場街	0.08~0.10 (0.09)	0.07~0.10 (0.09)	0.06~0.13 (0.10)

注) () は平均値

表 4-2-2 初期地表面残存負荷量(Ps0) (単位：kg/ha)

工種	用途	BOD	COD	SS
屋根		0.2	0.3	2.2
地表面	住宅地	3	7	10
地表面	商店街	12	16	24
地表面	工場街	5	8	10

限界降雨強度(rc)については、一般的に 2mm が用いられる⁷⁾が、JAMSOOP では別の値も設定可能である。また、補給係数は 2~3 日で 90%の回復が進むとの知見⁸⁾より、通常は 2 日間で Ps0 となる補給係数として設定する。

(2) T-P, T-N の場合

上記のように既往の研究が十分でない T-P と T-N の地表面パラメータについては、原則として解析対象地点における水質調査結果に基づき設定する。

以下に調査結果に基づく地表面パラメータの設定例を示す。

C 市で行われた分流雨水地区 (図 4-2-1) での調査に基づき、地表面残存負荷流出係数(K)と初期地表面残存負荷量 (P_{s0}) を設定した。

C 市における分流雨水の調査は 2 回実施されており、その際の流出負荷は図 4-2-2, 3, 表 4-2-3 に示すとおりである。

ここでは総流出負荷量が多い降雨 RC1 での総流出負荷量が、C-3 地点の集水エリア 81.98ha (不浸透面積率: 0.20) に残存していた T-P および T-N の総負荷量であると仮定し、初期地表面残存負荷量 (P_{s0}) を算定した。地表面残存負荷流出係数(K)については、総流出負荷量から逆算し設定した (表 4-2-4)。

限界降雨強度(r_c)については、BOD 等と同様に、一般的に用いられる 2mm とした⁷⁾。

また、補給係数についても、BOD 等と同様に 2~3 日で 90%の回復が進むとの知見⁸⁾より、2日間で P_{s0} となる補給係数とした。

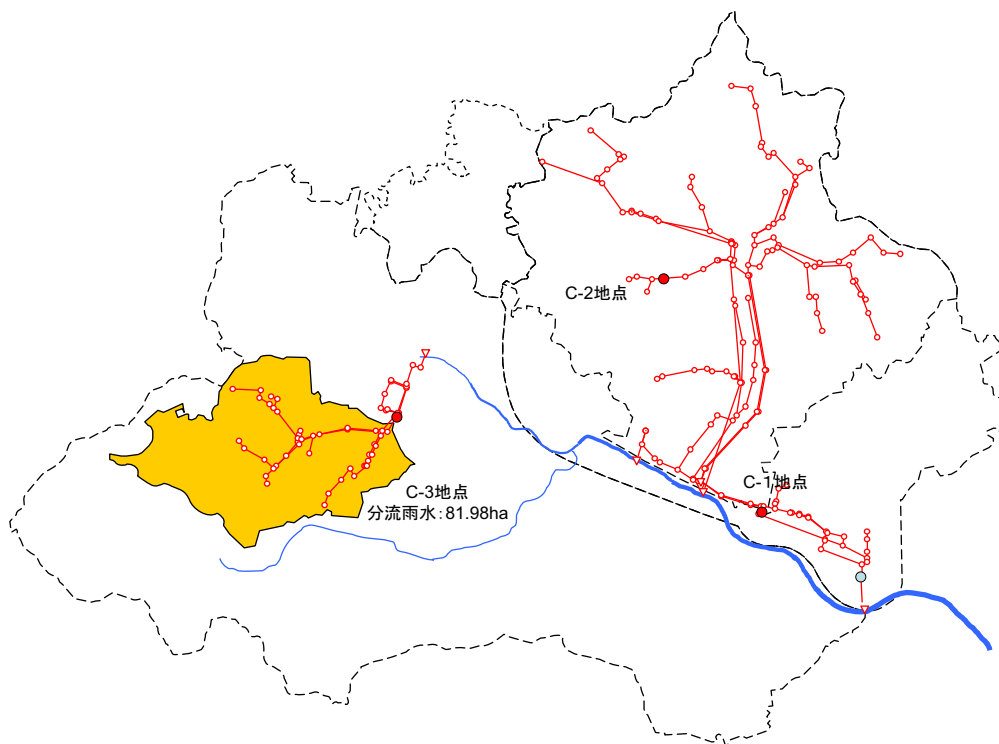


図 4-2-1 C 市分流雨水調査流域

表 4-2-3 C 市分流雨水における T-P,T-N 雨天時総流出負荷量

	総負荷量 (kg)		最大負荷量 (g/s)		最大降雨強度 (mm/hr)	汚濁負荷流出 対象時間
	T-P	T-N	T-P	T-N		
降雨 RC1	5.74	48.06	0.7	5.6	42.0	11 時間 (8:05-19:00)
降雨 RC2	2.57	27.3	0.1	0.5	18.0	23 時間 (13:10-12:00)

表 4-2-4 C 市分流雨水から求まる地表面のパラメータ

		T-P	T-N
総負荷量 (kg)		5.74	48.06
初期地表面残存負荷量 Ps0 (kg/ha)	総負荷量 /(81.98ha×0.20)	0.350	2.931
地表面残存負荷流出係数 K (1/mm)	$K = \text{最大負荷} \times 3.6 / (\text{Ps} \cdot (\text{最大降雨} - \text{rc}) \cdot A \cdot \text{imp})$	0.011	0.010
補給係数 a (kg/s/ha)	$\text{Ps0} / 48\text{hr}$ /(81.98ha×0.20)	2.026×10^{-6}	1.696×10^{-5}

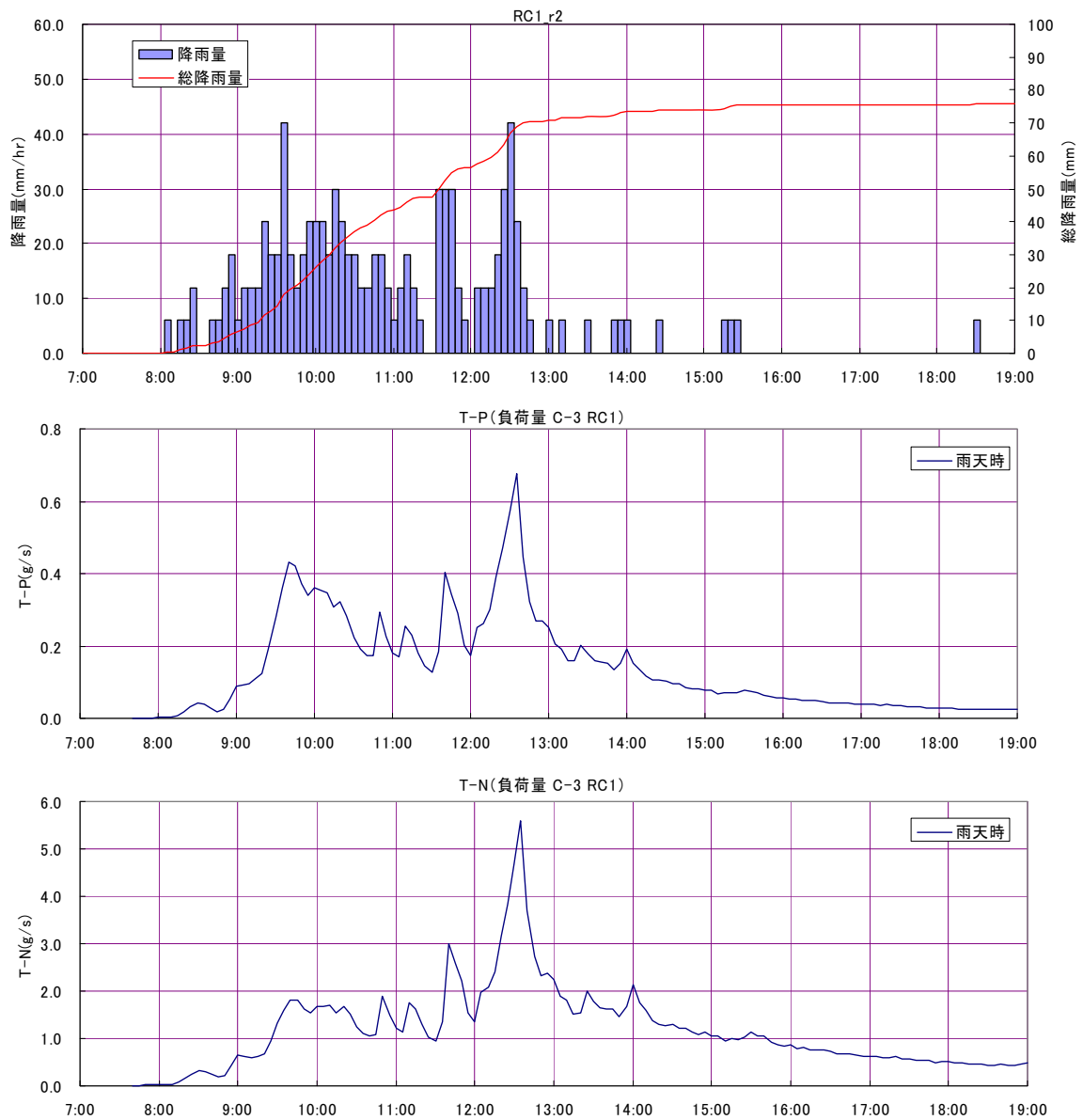


図 4-2-2 C 市 C-3 地点 (分流雨水) の流出汚濁負荷 降雨 RC1

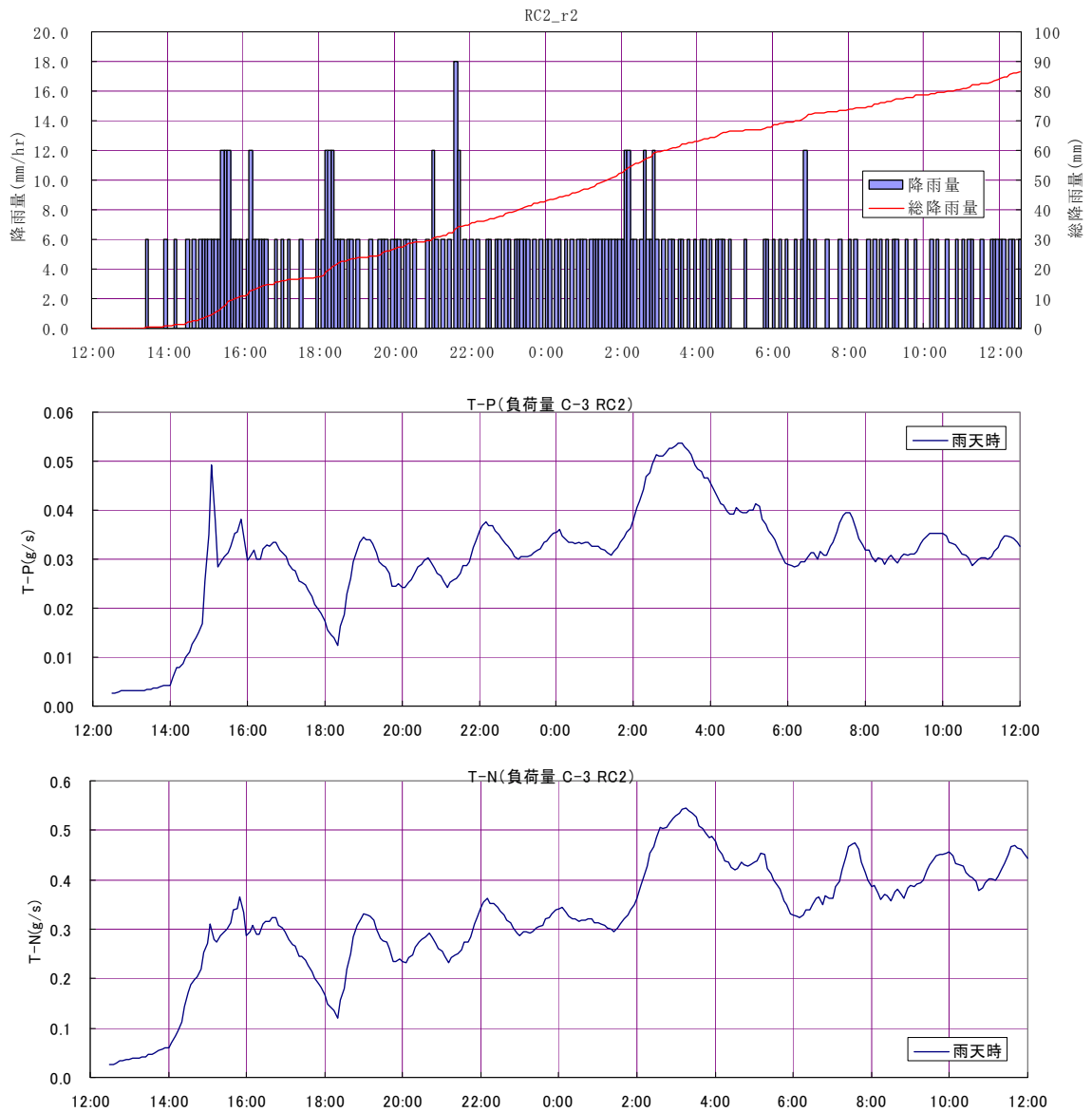


図 4-2-3 C 市 C-3 地点 (分流雨水) の流出汚濁負荷 降雨 RC2

5. ケーススタディ

5. ケーススタディ

ここでは、分布型汚濁負荷モデルの活用事例として、下記の仮想モデル地区に合流改善対策（浸透，貯留，分流化）対策を行い、分布型汚濁負荷モデルで評価した場合の例を示す。

5-1. 仮想モデル地区の設定

ケーススタディ対象とする仮想モデル地区は流域面積約 130ha とし、流域内に非対策地区と対策地区を想定して各地区のモデル構築を行った。

- (1) 流域諸元
- (2) 対象降雨
- (3) 汚濁負荷パラメータ

(1) 流域諸元

仮想モデル地区は流域内に分水人孔及びバイパス管を有する流域(図 5-1-1, 全体 130ha)とした。

また、現状では合流式で流域全体が表 5-1-1 の諸元で整備済みであると想定した。

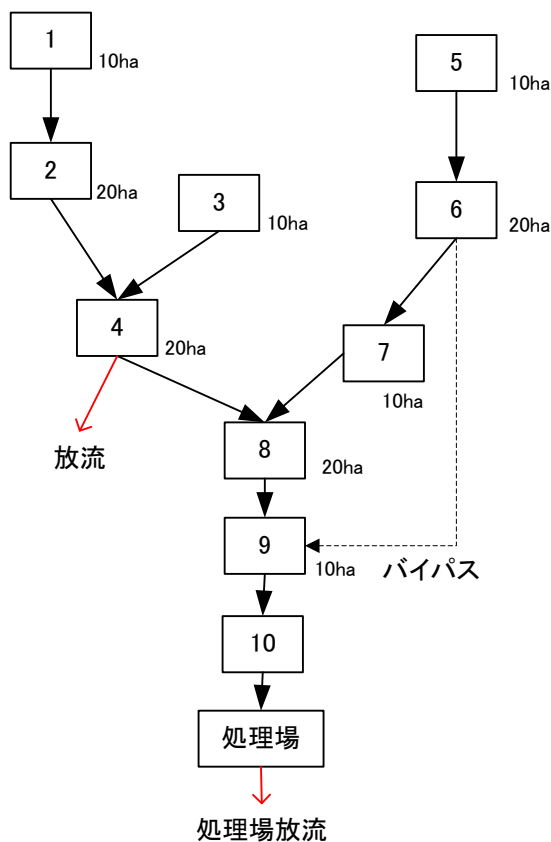


図 5-1-1 仮想モデル地区 (130ha)
(流域 10 は、流域 9 と処理場をつなぐための仮想流域)

表 5-1-1 モデル地区想定諸元

項目	単位	諸元	備考
不浸透面積率 (Imp)	—	0.5	
浸透域浸透能 (Fc)	mm/hr	10	
浸透域凹地割合 (Ap)	%	80	
浸透域凹地貯留高 (Dp)	mm	6	
不浸透域凹地割合 (Ai)	%	60	
不浸透域凹地貯留高 (Di)	mm	2	
遮集倍率	Q	3	
晴天時平均汚水量	m ³ /s/ha	0.001	
流達時間	分	5	各路線で統一
SQカーブ式：K	—	0.08	各路線で統一
SQカーブ式：P	—	0.7	

(2) 対象降雨

開発したソフトウェア JAMSOOP ver.1.1.0 では一度に複数降雨を対象に解析を行うことが可能である。ここでは以下の 12 降雨をモデル降雨として解析および集計を行うこととした。

なお、降雨データは仮想的に作成したものである。

表 5-1-2 対象降雨諸元

降雨番号	総降雨量 (mm)	最大降雨強度 (mm/10分)	降雨継続時間 (分)
1	8.5	2.0	120
2	2.5	1.0	30
3	9.5	2.0	140
4	22.5	1.0	1020
5	8.0	3.0	150
6	6.0	2.0	110
7	41.5	5.5	720
8	13.0	2.0	180
9	1.5	0.5	30
10	2.5	0.5	360
11	6.0	2.0	60
12	11.0	2.0	140

(3) 汚濁負荷パラメータ

① 初期管内堆積負荷量 Pp0 および負荷流出係数 C 等

分布型汚濁負荷モデルに検討に必要なパラメータの設定は、晴天時流量および水質を基に、単位面積あたりの日発生負荷量を算定して設定する。

○初期管内堆積負荷量(Pp0)

Pp0 は流域毎に、各地点における日発生負荷量の 0.5 倍とする。

$$\text{日発生負荷量(kg)} = \text{晴天時流量(m}^3\text{/s)} \times \text{平均水質(mg/L=g/m}^3\text{)} \times 86400\text{(s)} \div 1000$$

各水質項目の平均水質は以下のとおり設定した。

表 5-1-3 平均水質の設定(mg/L)

	BOD	COD	SS	T-P	T-N
平均水質(mg/L)	200	150	180	5	30

○負荷流出係数 (C)

負荷流出係数 C は以下の式により、モデル内で自動的に算出される。

$$C = \text{晴天時平均水質} \times 1.2 / Pp0^2 \quad \dots \text{(SS 以外)}$$

$$C = \text{晴天時平均水質} / (Pp0 \cdot \text{日平均流量}) \quad \dots \text{(SS)}$$

○限界流量(qc)

限界流量は各流域における日平均汚水量の 1/10 = 0.0001m³/s/ha とした。

③ 地表面パラメータ

地表面パラメータは表 5-2-3 に示すよう設定した。

表 5-1-4 地表面モデルパラメータ

	BOD	COD	SS	T-P	T-N
初期地表面残存負荷量 Ps0 (kg/ha)	5.0	4.0	5.0	0.35	2.9
地表面残存負荷流出係数 K (1/mm)	0.1	0.15	0.4	0.35	0.01
補給係数 a (g/s/ha)	0.029	0.023	0.029	0.002	0.017
限界降雨強度 rc (mm)	2.0	2.0	2.0	2.0	2.0

5-2. シミュレーション結果

ここでは、対策施設なしの場合のシミュレーション結果を示す。

- (1) 降雨と流量、汚濁負荷量の関係の例
- (2) 水質項目別の解析結果の例
- (3) 処理場計算
- (4) 雨水吐き口集計

(1) 降雨と流量、汚濁負荷量の関係の例

総降雨量の異なる降雨1、4、7を対象とした、最下流である流域10からの流量および汚濁負荷量(処理場への流量及び汚濁負荷量)の流出解析結果を例として示す。

(なお、降雨量と流量・負荷量の同一のグラフ上への表示は JAMSOOP 本体のグラフ表示ツールでは対応していないため、以下に示すグラフは、解析結果をエクセルで処理したものである。)

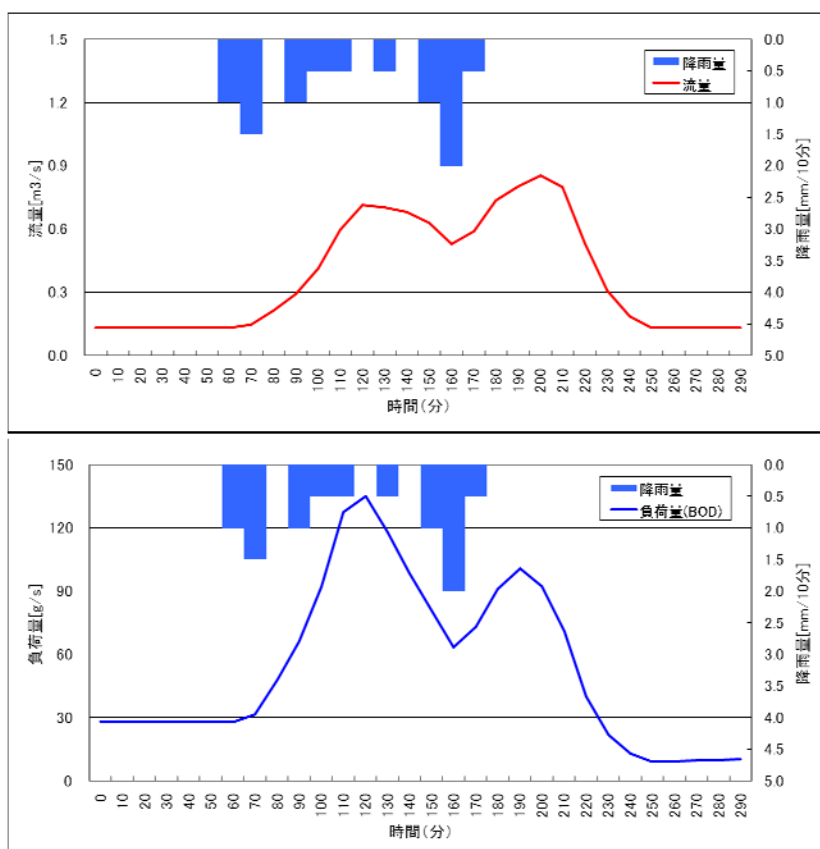


図 5-2-1 流域 10 における降雨 1 での解析結果 (流量、BOD)

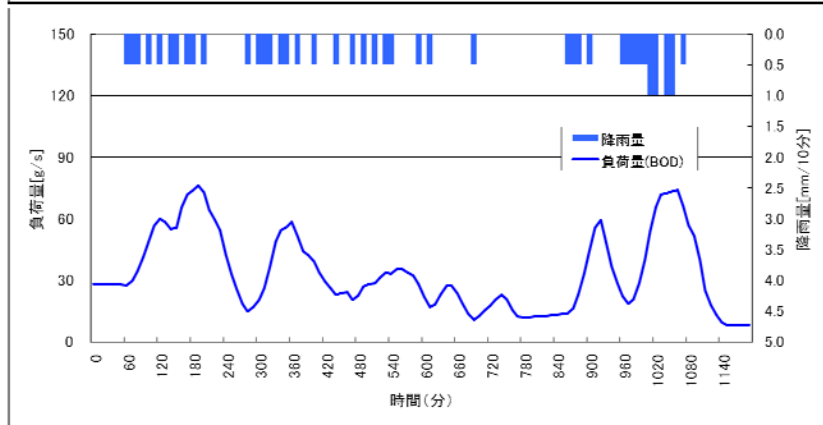
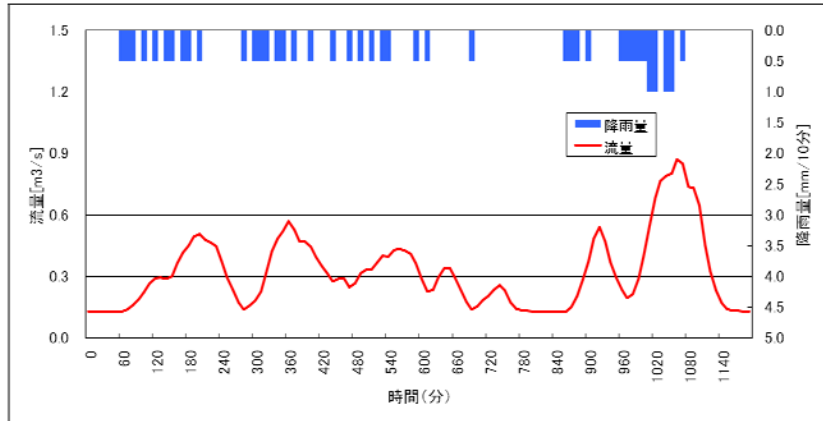


図 5-2-2 流域 10 における降雨 4 での解析結果 (流量、BOD)

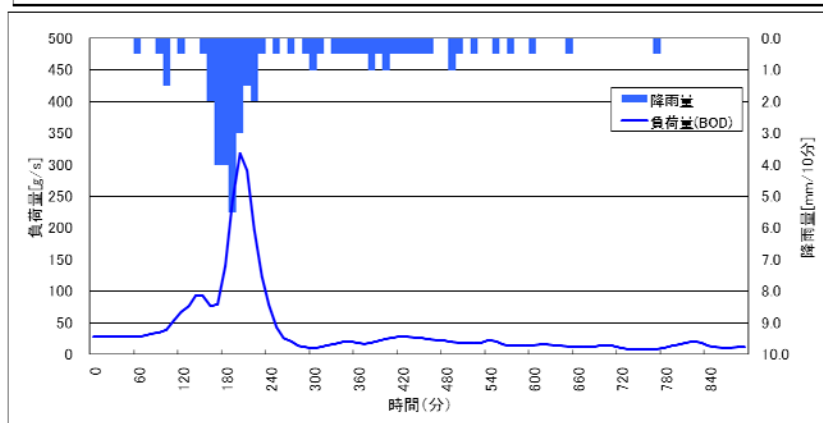
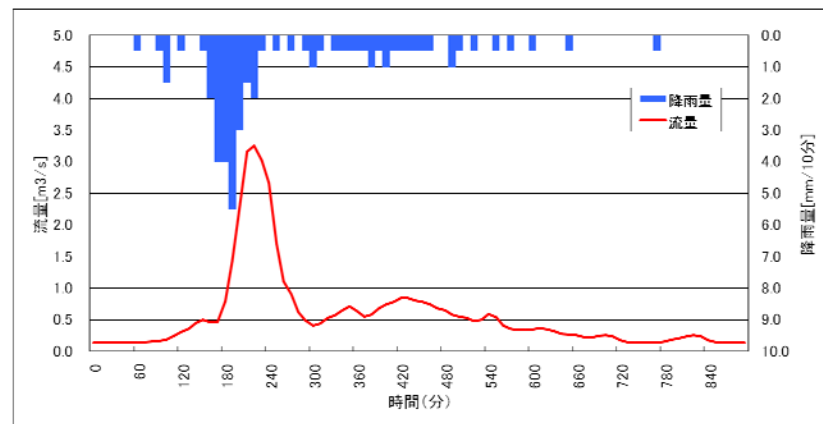


図 5-2-3 流域 10 における降雨 7 での解析結果 (流量、BOD)

(2) 水質項目別の解析結果の例

降雨 4 に対する流域 10 からの流量および水質 5 項目の汚濁負荷量の解析結果を例として示す。BOD、COD、T-N については概ね同様の流出波形となっているが、SS、T-P は波形が異なり、降雨後半のピークに汚濁負荷の流出が集中している。これは、SS は管内負荷流出のモデル式が異なること、T-P は平均水質が他の水質項目に対し小さいため、流出係数が大きくなることに由来する。

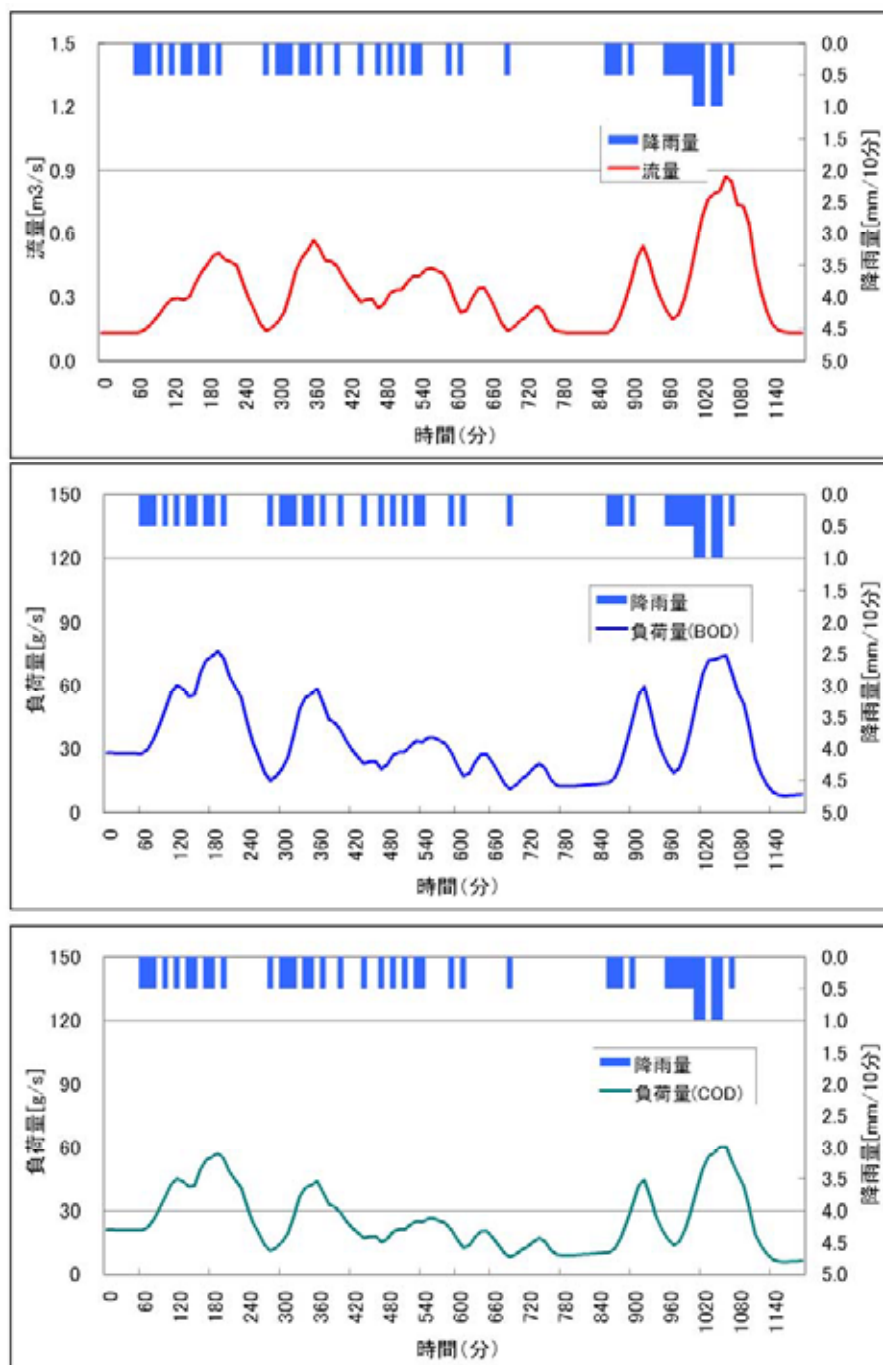


図 5-2-4 流域 10 での各項目の解析結果（上から流量、BOD、COD）

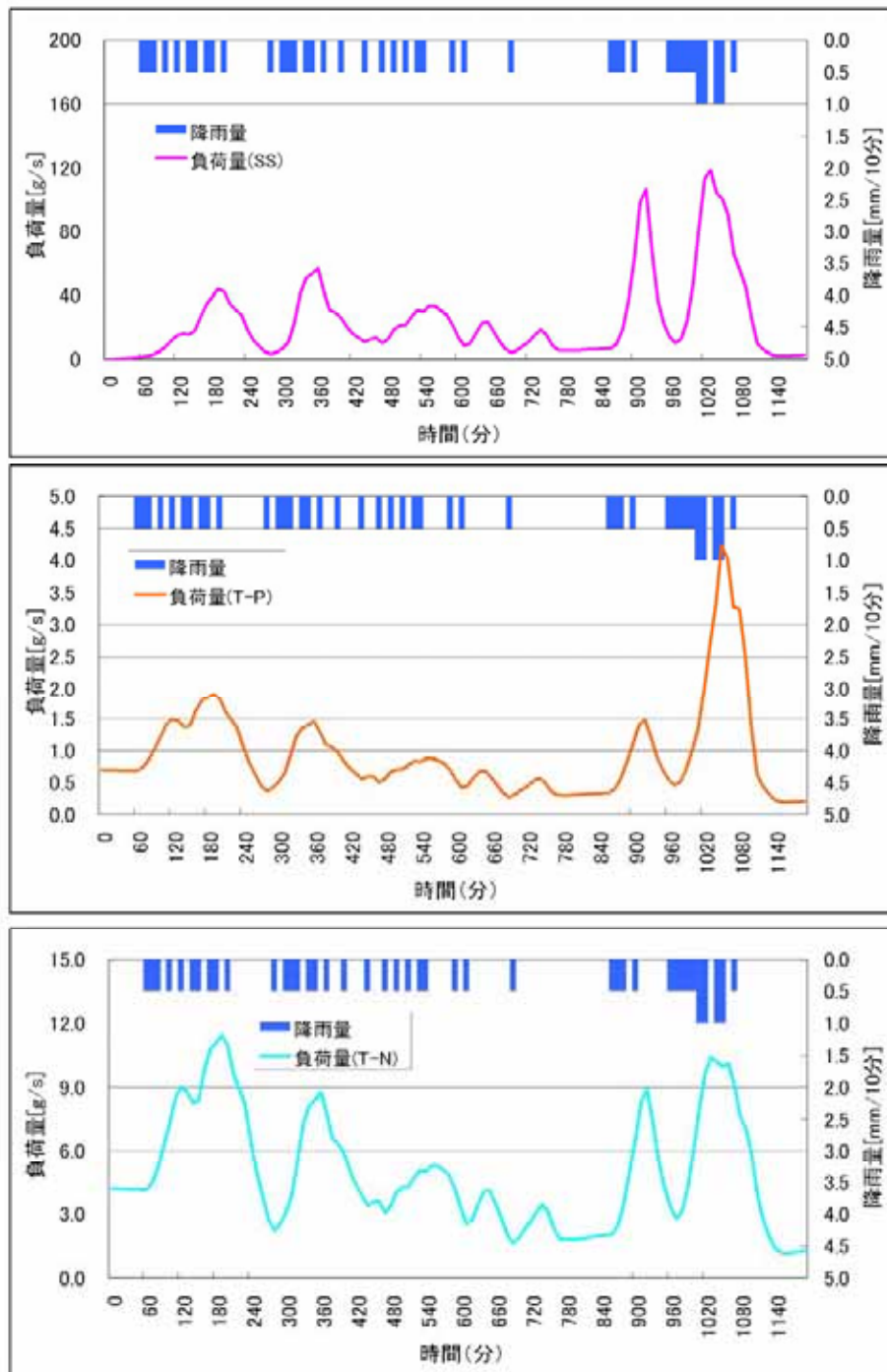


図 5-2-5 流域 10 での各項目の解析結果 (上から SS、T-P、T-N)

(3) 処理場計算

降雨 1、4、7 における処理場の計算結果グラフを図 5-2-6～図 5-2-8 に、また 12 降雨の集計結果を表 5-2-1 に示す。図中の汚濁負荷量は、処理後の BOD を示している。

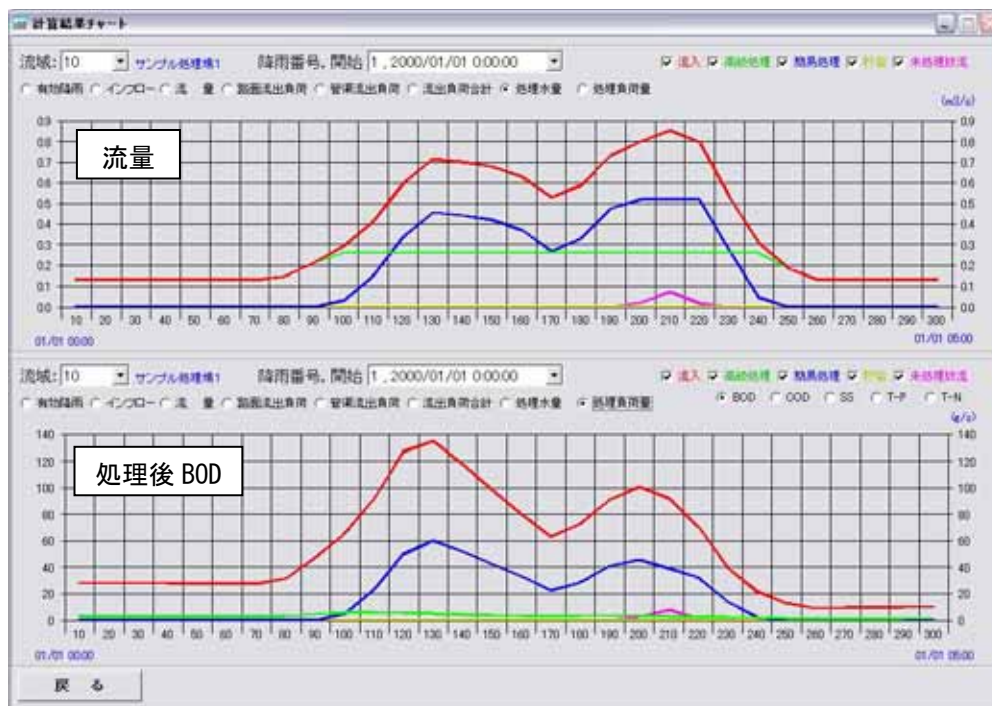


図 5-2-6 処理場計算結果（降雨 1）

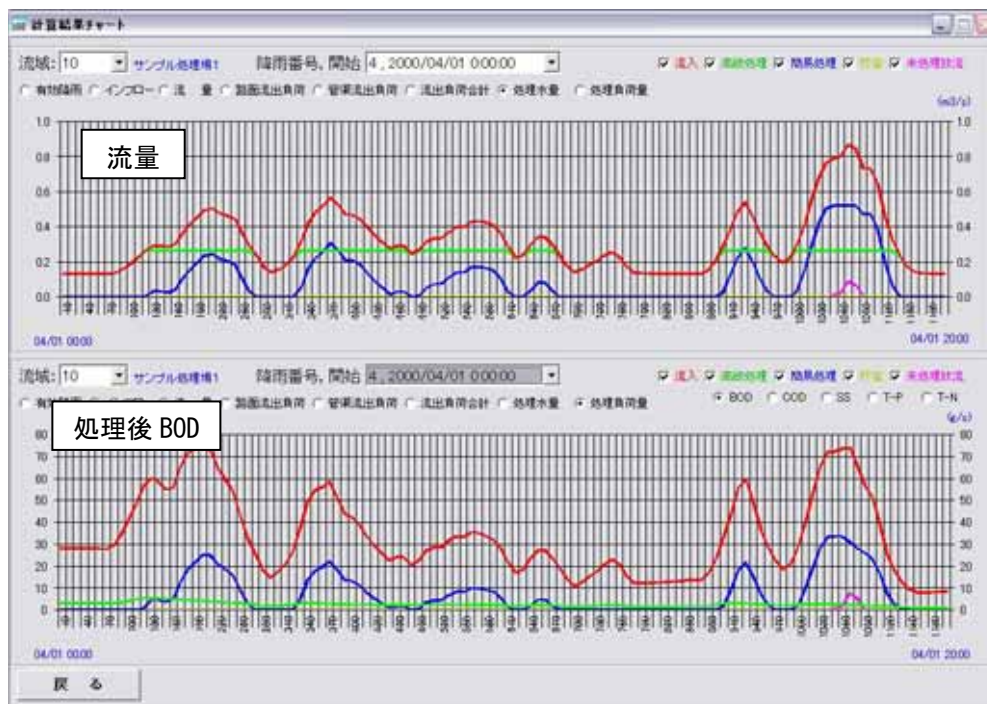


図 5-2-7 処理場計算結果（降雨 4）

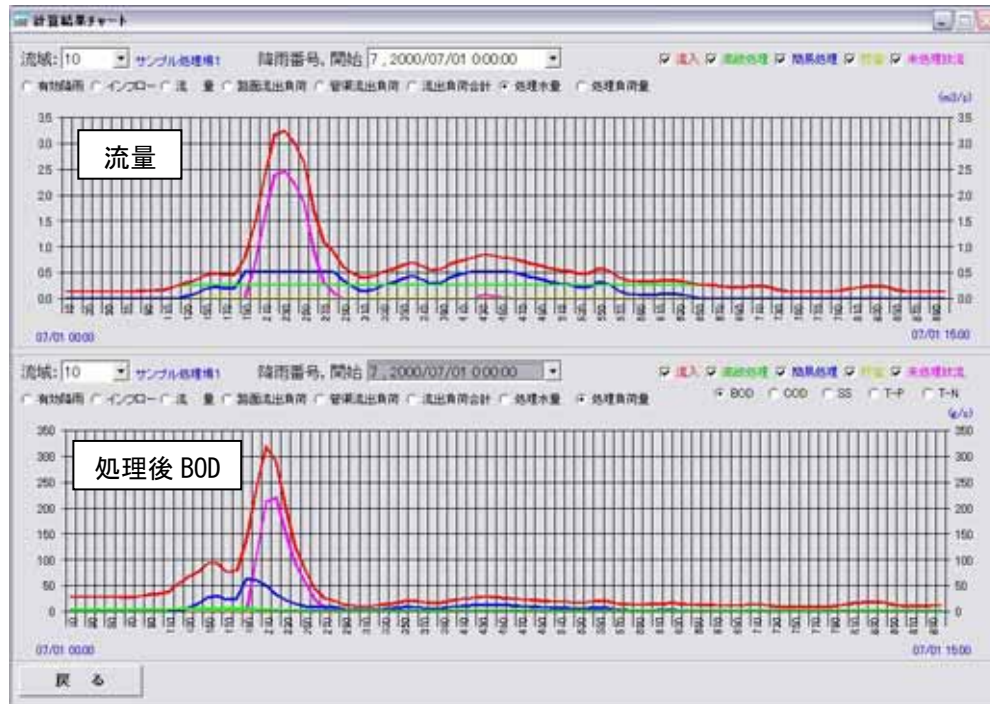


図 5-2-8 処理場計算結果（降雨 7）

表 5-2-1 降雨 1～12 を対象とした処理場における計算集計

	高級処理	簡易処理	直接放流	計
水量(m ³)	63,049	36,924	10,512	110,485
処理前負荷量(kg)	8,141	3,974	925	13,040
処理後負荷量(kg)	814	2,782	925	4,521
放流回数	-	-	9	

(4) 雨水吐き口集計

流域 4 の雨水吐き口からの未処理放流については以下のとおりである。

表 5-2-2 降雨 1～12 を対象とした雨水吐き口における計算集計

直接放流量	直接放流負荷量	放流回数
m ³	kg	回
11,325	952	9

5-3. 浸透施設の整備

ここでは、対象流域に浸透施設の整備を行った場合の例を示す。

- (1) 雨水浸透施設
- (2) 浸透能力の設定
- (3) 降雨と流量、汚濁負荷量の関係の例
- (4) 処理場集計
- (5) 雨水吐き口集計

(1) 雨水浸透施設

一般に雨水浸透施設は、側面や底面に浸透孔を有する浸透ますや浸透トレンチ、空隙率の高いブロック等からなる透水性舗装等がある。

一般に浸透施設を整備することにより、整備した地区からの流出量は図 5-3-2 に示すように、浸透能力分が差し引かれたものとなる。

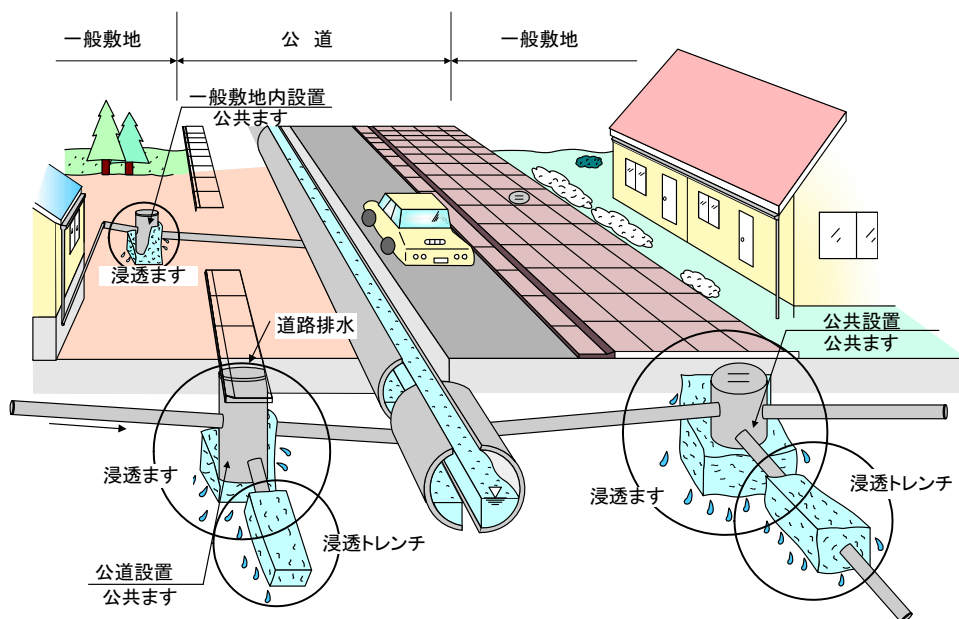


図 5-3-1 浸透施設の整備イメージ 9)

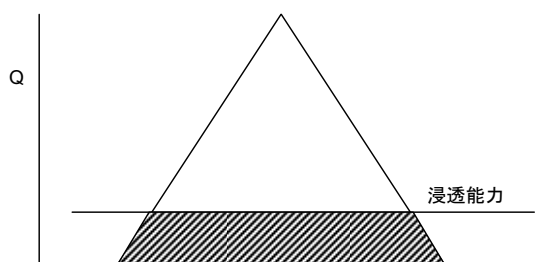


図 5-3-2 浸透による効果のイメージ

(2) 浸透能力の設定

浸透施設の設置により、不浸透域率が 0.5 から 0.4 へ、全体に対して 1 割減少すると想定した。浸透能 10mm/hr の浸透域が 1 割増加することで、流域平均で 1mm/hr の浸透対策となる。なお、その他の設定は変更していない。

表 5-3-1 モデル地区想定諸元 (浸透)

項目	単位	諸元	諸元(浸透)	備考
不浸透面積率 (Imp)	—	0.5	0.4	
浸透域浸透能 (Fc)	mm/hr	10	10	
浸透域凹地割合 (Ap)	%	80	80	
浸透域凹地貯留高 (Dp)	mm	6	6	
不浸透域凹地割合 (Ai)	%	60	60	
不浸透域凹地貯留高 (Di)	mm	2	2	

(3) 降雨と流量、汚濁負荷量の関係の例

降雨 1、4、7 を対象とした、最下流である流域 10 からの流量および BOD の流出解析結果について、対策なしのケースとの比較を例として示す。流量・BOD の両者とも、対策なしよりも減少している。

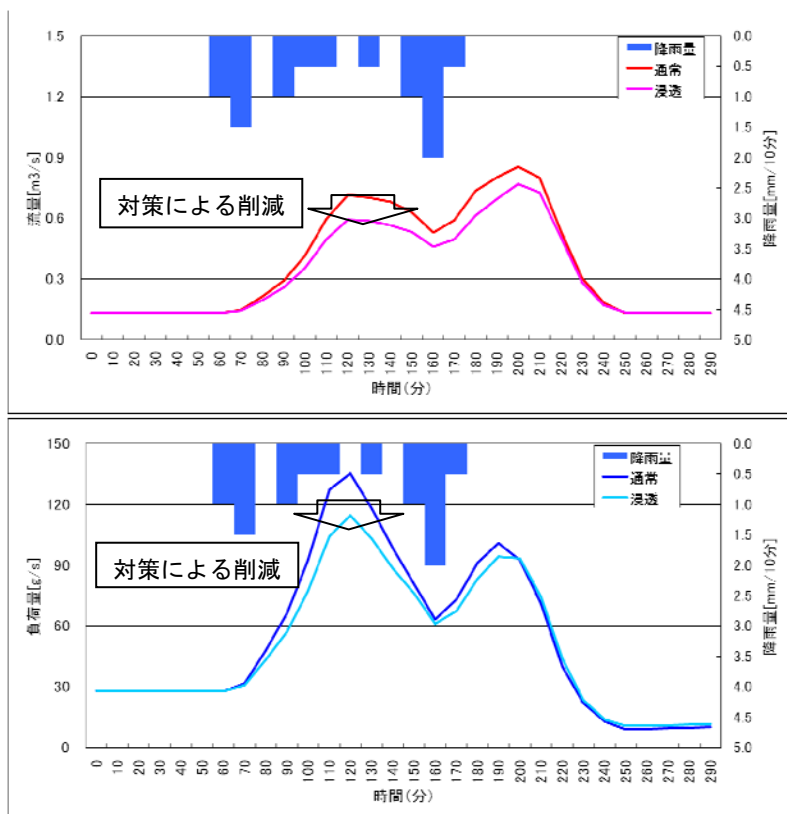


図 5-3-3 流域 10 における降雨 1 の解析結果 (流量、BOD)

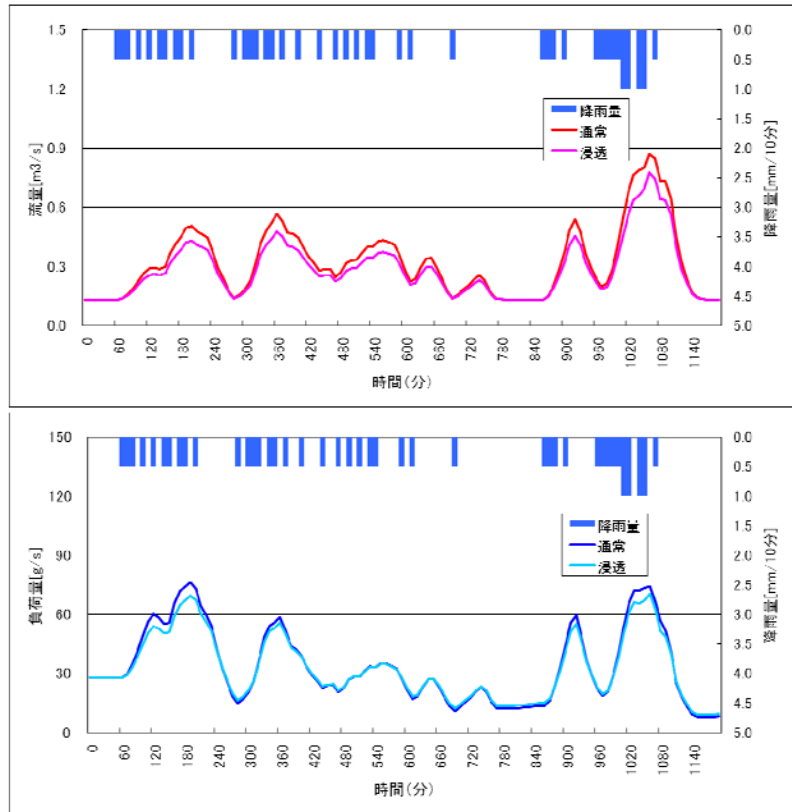


図 5-3-4 流域 10 における降雨 4 の解析結果 (流量、BOD)

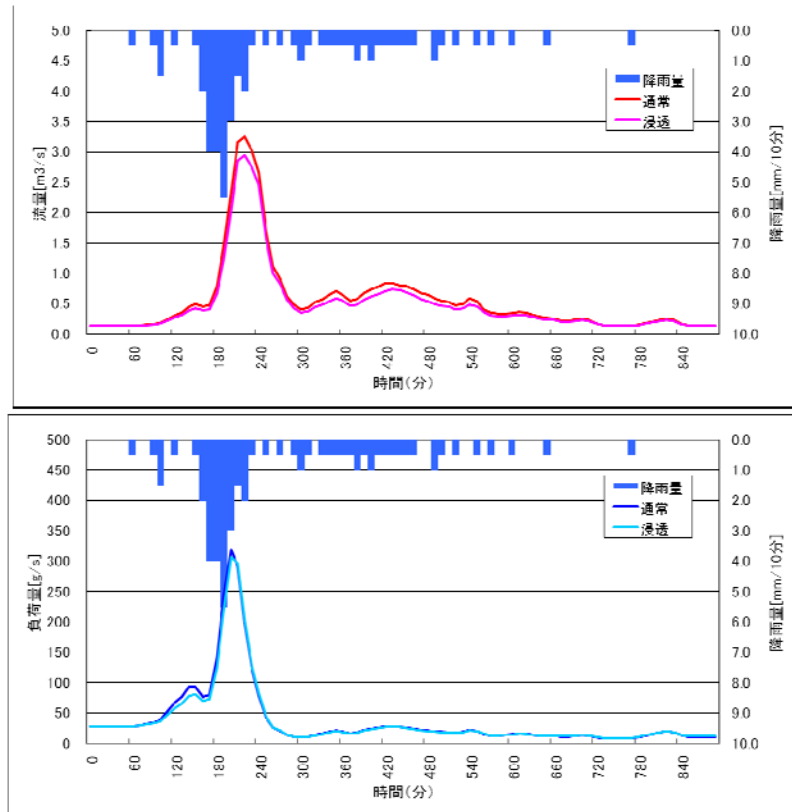


図 5-3-5 流域 10 における降雨 7 の解析結果 (流量、BOD)

(4) 処理場集計

降雨1、4、7を対象とした処理場の流量および汚濁負荷量の流出解析結果について、対策なしのケースとの比較を例として示す。また12降雨の集計結果の比較を表5-3-2に示す。図中、汚濁負荷量は処理後の汚濁負荷量である。

降雨期間全体に対する浸透施設のベースカット効果により、流量・汚濁負荷量とも全体的にピークが低下しており、また対策前は降雨1、4でわずかに発生していた未処理放流が、浸透施策により発生しなくなっている。

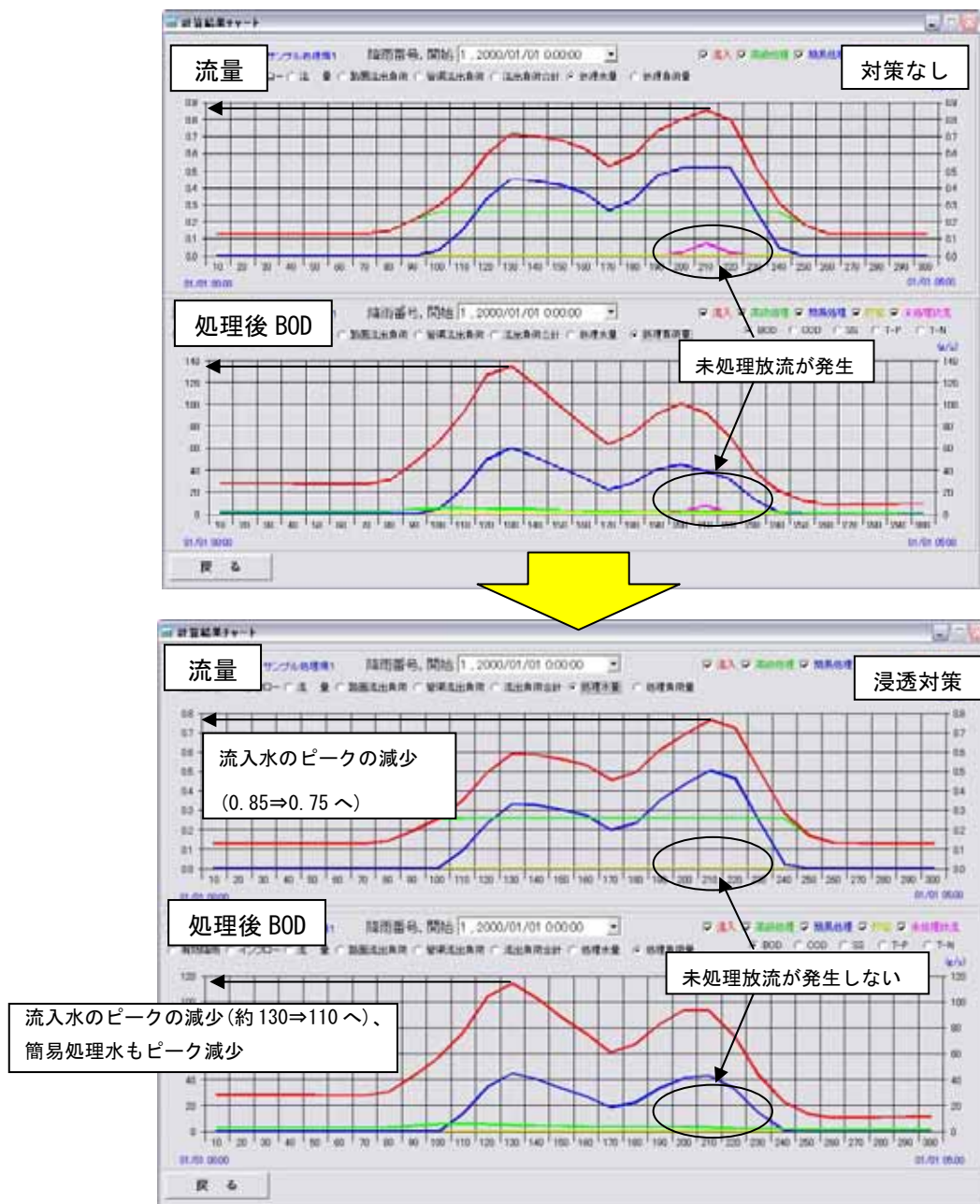


図 5-3-6 処理場計算結果（降雨1）（上：対策なし、下：浸透対策）

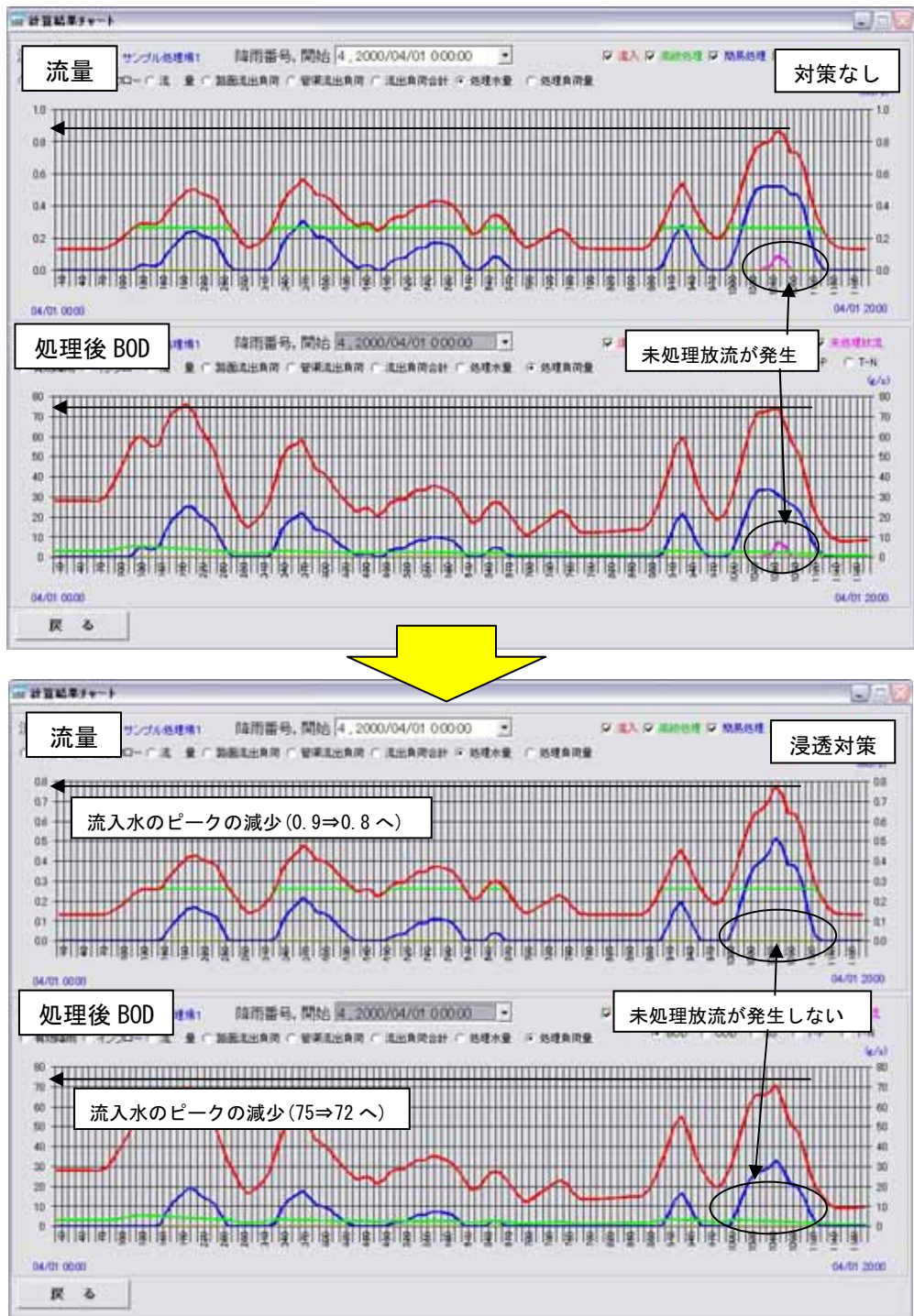


図 5-3-7 処理場計算結果 (降雨4) (上: 対策なし、下: 浸透対策)

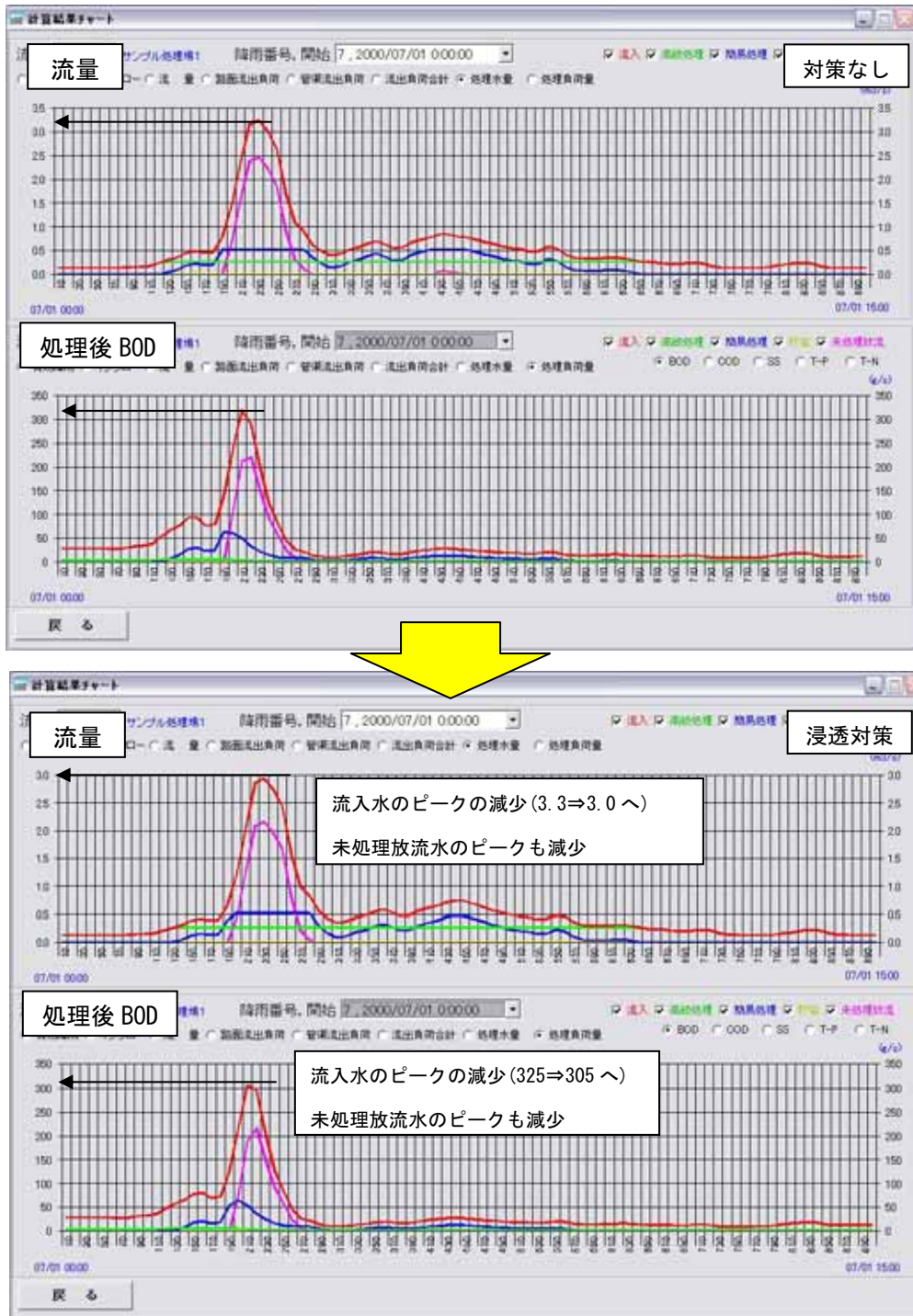


図 5-3-8 処理場計算結果 (降雨7) (上: 対策なし、下: 浸透対策)

表 5-3-2 降雨 1～12 を対象とした処理場における計算集計

	検討ケース	高級処理	簡易処理	直接放流	計
水量(m ³)	通常	63,049	36,924	10,512	110,485
	浸透	61,977	29,659	7,438	99,073
処理前負荷量(kg)	通常	8,141	3,974	925	13,040
	浸透	8,388	3,416	640	12,444
処理後負荷量(kg)	通常	814	2,782	925	4,521
	浸透	839	2,391	640	3,870
放流回数	通常	-	-	9	
	浸透	-	-	7	

表 5-3-2 に示すとおり、流域平均 1mm/hr の浸透施策を実施することで BOD 汚濁負荷の発生が 4,521kg から 3,870kg へ、約 15%削減された。

(5) 雨水吐き口集計

流域 4 の雨水吐きにおいては、放流回数は 9 回で対策なしの場合と同じであったが、放流量は 11,325m³ から 8,184m³ へ、BOD 放流負荷量は 952kg から 687kg へ、それぞれ約 3 割減となった。

表 5-3-3 降雨 1～12 を対象とした雨水吐き口における計算集計

	直接放流量	直接放流負荷量	放流回数
	m ³	kg	回
通常	11,325	952	9
浸透	8,184	687	9

5-4. 滞水池の整備

ここでは、雨水吐きのある流域 4 および処理場内に滞水池を設置した場合の例を示す。

- (1) 雨水滞水池
- (2) 滞水池の設定
- (3) 降雨と流量、汚濁負荷量の関係の例
- (4) 処理場集計
- (5) 雨水吐き口集計

(1) 雨水滞水池

雨水滞水池は、雨水吐き室、ポンプ場からの未処理下水あるいは遮集雨水の一部を貯留または沈殿し、貯留した雨天時下水を降雨終了後、処理場等に送水して処理を行うことにより、公共用水域等へ排水される雨天時負荷量等の削減を図る施設である¹⁾。

一般に雨水滞水池の形式は、流入開始条件と放流条件の違いにより、表 5-4-1 に示す形式に分類される。

(2) 滞水池の設定

以下に示す例では形式Ⅱ、Ⅲに相当するタイプの滞水池を想定する。

雨水吐き口のある流域 4 および処理場において、それぞれ容量 1000m³の滞水池を設置し、流域 4 では 3Q 以上の未処理放流水を対象に貯留(Ⅱ型)、処理場では流量が 1Q を超えると貯留を開始(Ⅲ型)する設定とした。

これは流域全体に対し、約 1.5mm 分の貯留高に相当する。

(3) 降雨と流量、汚濁負荷量の関係の例

貯留対策は未処理放流の越流後の下水や処理場流入後の下水を対象とした対策であるため、流域 10 における流量、汚濁負荷は対策なしのケースと同じである。

表 5-4-1 雨水滞水池の形式 1)

形式	ハイドログラフとの関係 Q_2 = 計画時間最大汚水量 Q_3 = 遮集管きよ容量	説明	特徴	評価
I		<ol style="list-style-type: none"> 1. 遮集管きよ容量を上回ると雨水滞水池に流入する。 2. 満水になっても、遮集管きよ容量を上回る水量は、雨水滞水池に流入し、流入水量と等しい水量が放流される。 	<ol style="list-style-type: none"> 1. 降雨中は沈殿作用がある。 2. 貯留下水を高級処理することが可能である。 	<ol style="list-style-type: none"> 1. 簡易処理量が存在するため、小降雨時に雨水滞水池の効果が小さい。 2. 満水になると、初期雨水も流出する懸念がある。
II		<ol style="list-style-type: none"> 1. 流入はIと同じ。 2. 雨水滞水池が満水になると、それ以後の流入下水は、公共用水域に直接放流される。 	<ol style="list-style-type: none"> 1. 貯留下水を高級処理することが可能である。 	<ol style="list-style-type: none"> 1. 簡易処理量が存在するため、小降雨時に雨水滞水池の効果が小さい。 2. 雨水滞水池の容量が大きいと、効果的である。 3. IIIと比較して未処理放流負荷量の削減効果が大きい。
III		<ol style="list-style-type: none"> 1. 雨天時合流下水が計画時間最大汚水量を上回ると、雨水滞水池に流入する。 2. 雨水滞水池が満水になるまでは、計画時間最大汚水量のみが処理場に送水される。 3. 満水以後はIIと同じ。 	<ol style="list-style-type: none"> 1. 雨水滞水池が満水になるまでには、処理場は簡易処理がなく、小降雨時の簡易処理量は減少する。 2. 貯留下水を高級処理することが可能である。 	<ol style="list-style-type: none"> 1. 使用頻度が高く、小降雨時に特に効果的となる。 2. IIと比較して、放流負荷量の削減効果が大きい。
IV		<ol style="list-style-type: none"> 1. 計画時間最大汚水量を上回り、遮集管きよ容量以下の遮集雨水量を雨水滞水池に流入させる。 2. 遮集管きよ容量を上回る雨天時合流下水は、直接放流される。 	<ol style="list-style-type: none"> 1. 簡易処理量が減少する。 2. 他の形式と比較して未処理下水水量が多い。 3. 貯留下水は、高級処理することが可能である。 	<ol style="list-style-type: none"> 1. 小降雨時にIIIと同じ効果がある。 2. 大きな降雨になると、上流のポンプ場や雨水吐き室からの未処理放流があるので、効果は比較的小さくなる。

(4) 処理場集計

降雨1、4、7を対象とした処理場の流量および汚濁負荷量の流出解析結果について、対策なしのケースとの比較を例として示す。また12降雨の集計結果の比較を表5-4-2に示す。図中、汚濁負荷量は処理後の汚濁負荷量である。

流域4において3Q以上の雨水を貯留することで、処理場へ流入する水量および汚濁負荷の初期ピークが低減している。また、処理場で1Q以上の雨水をさらに貯留することで、簡易処理水が削減され、全体として汚濁負荷の発生が削減される。

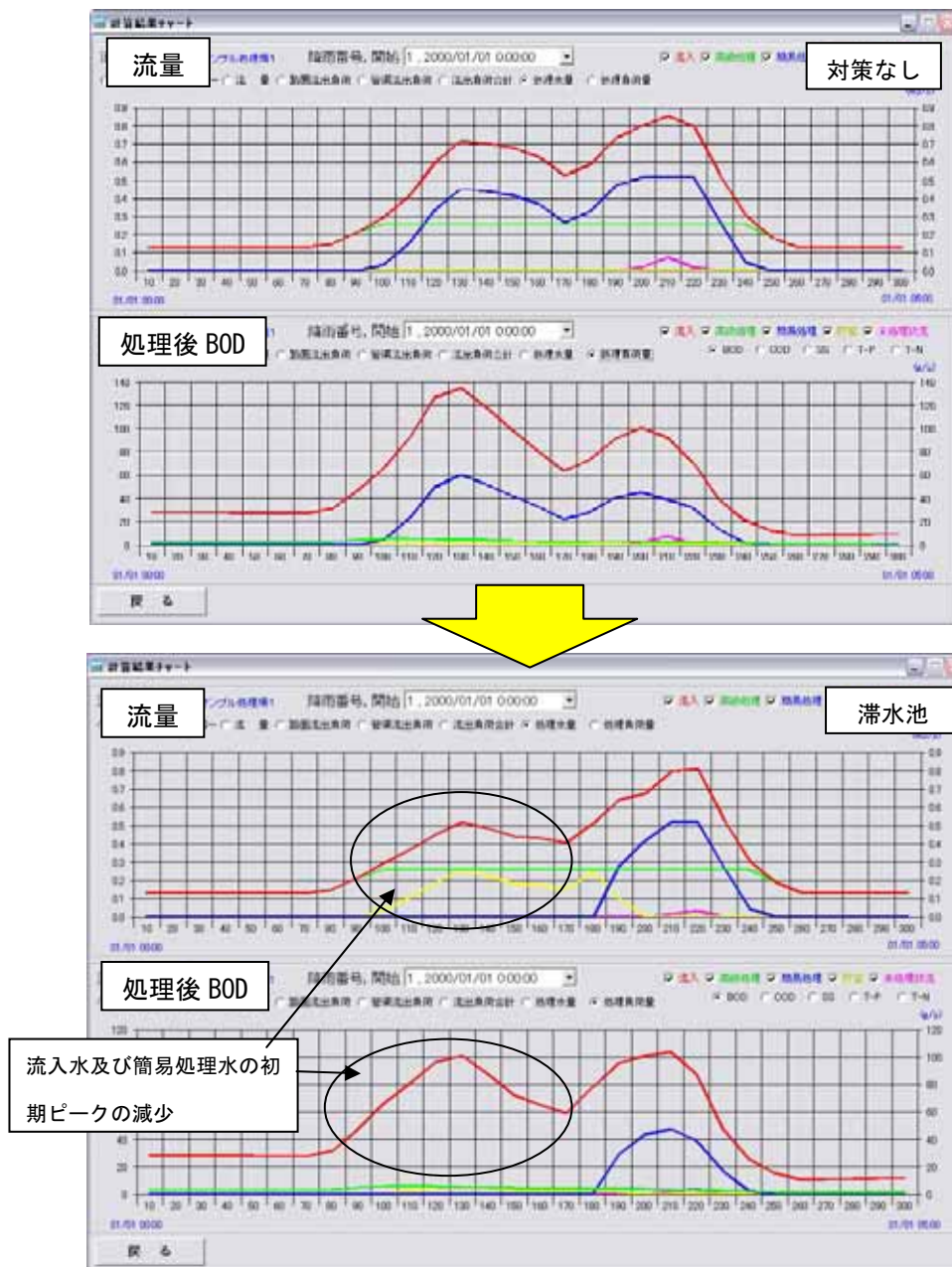


図 5-4-1 処理場計算結果 (降雨1) (上: 対策なし、下: 滞水池)

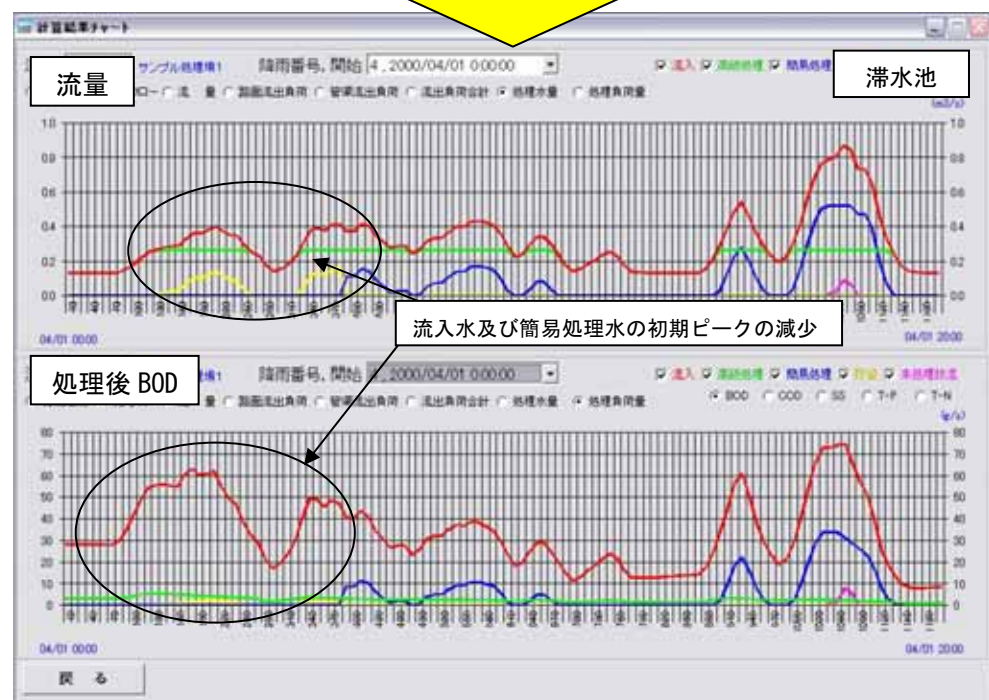
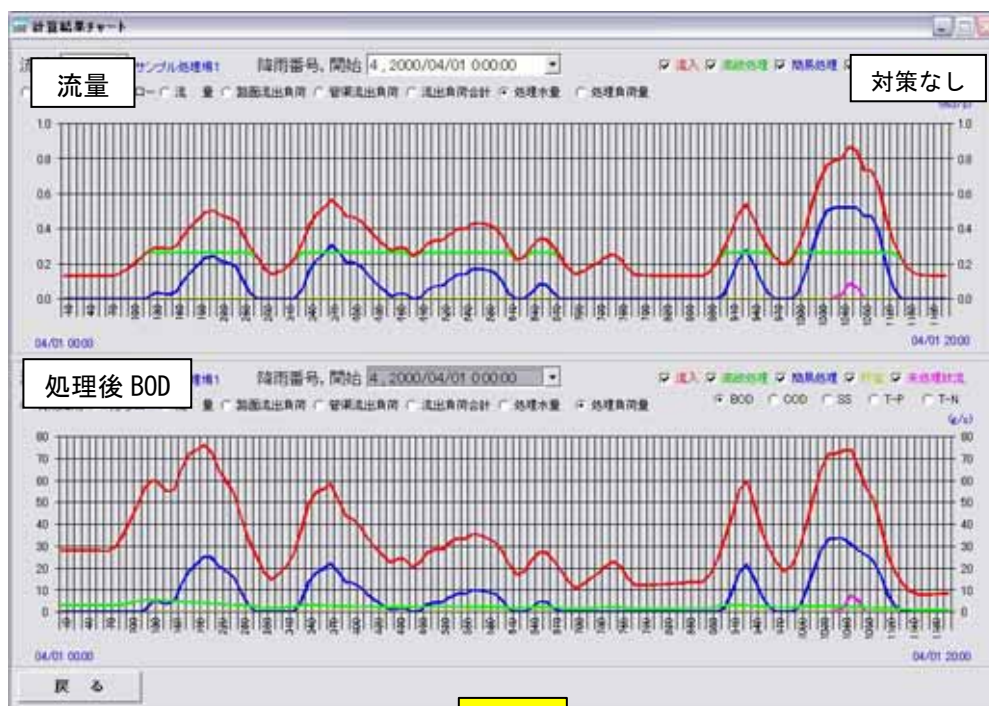


図 5-4-2 処理場計算結果（降雨4）（上：対策なし、下：滞水池）

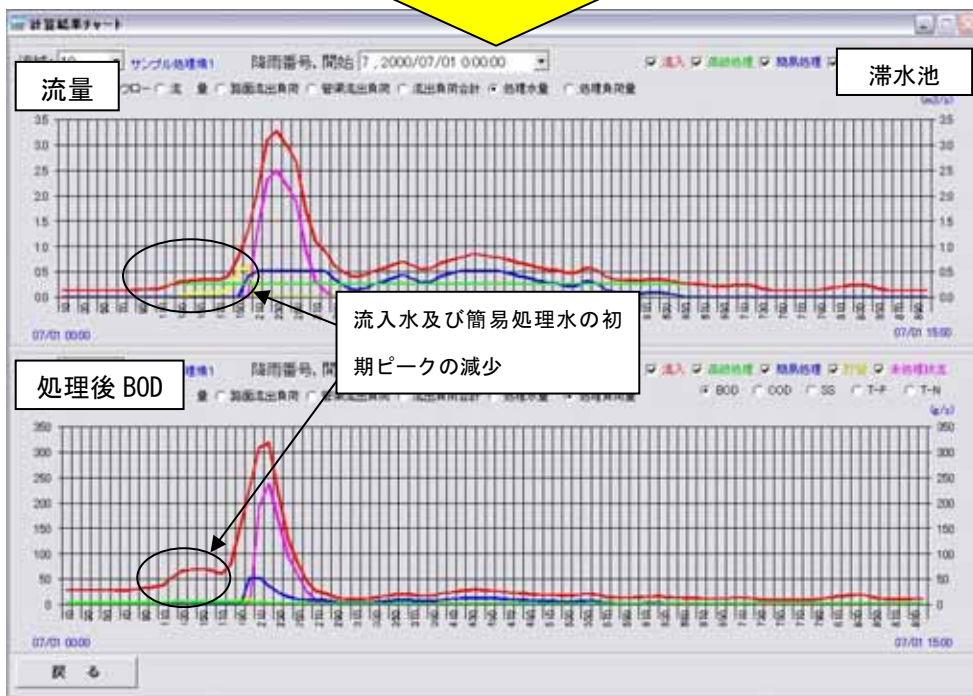
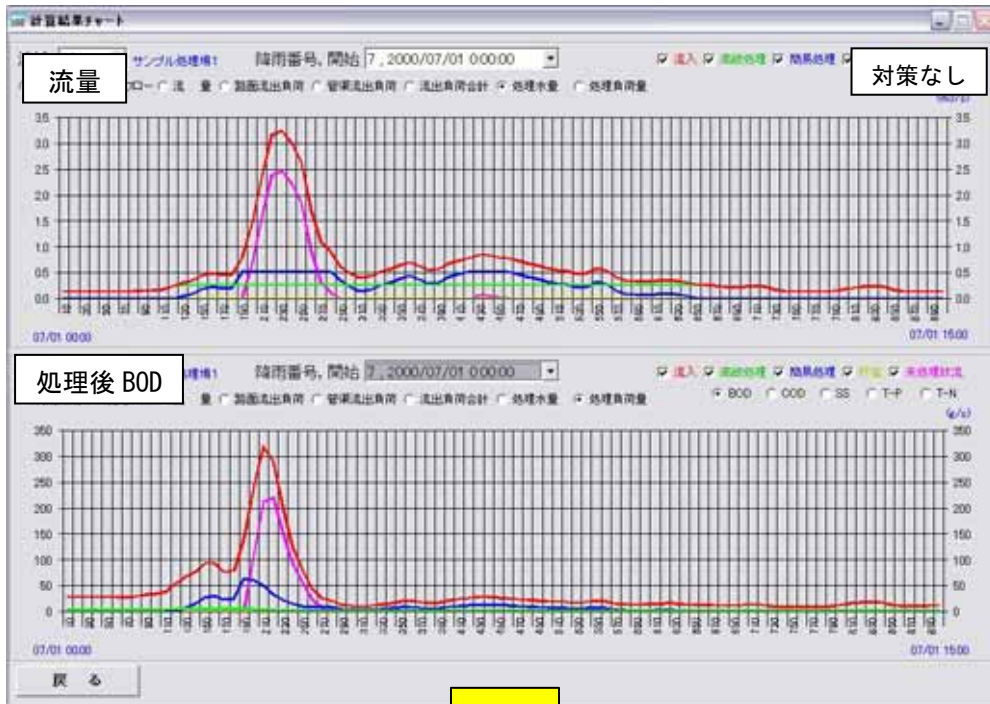


図 5-4-3 処理場計算結果（降雨 7）（上：対策なし、下：滞水池）

表 5-4-2 降雨 1～12 を対象とした処理場における計算集計

	検討ケース	高級処理	簡易処理	貯留 (カック内は 処理場滞水池)	直接放流	計
水量(m ³)	通常	63,049	36,924	-	10,512	110,485
	貯留	63,004	23,864	9,120	8,273	104,260
処理前負荷量(kg)	通常	8,141	3,974	-	925	13,040
	貯留	8,459	1,904	1,704	613	12,680
処理後負荷量(kg)	通常	814	2,782	-	925	4,521
	貯留	846	1,333	170	613	2,962
放流回数	通常	-	-	-	9	-
	貯留	-	-	-	6	-

表 5-4-2 に、前述の滞水池整備による雨天時放流負荷の削減効果を示す。表に示すとおり、約 1.5mm/hr の貯留施策を実施することで BOD 汚濁負荷の発生が 4,521kg から 2,962kg へ、約 34%削減された。

(5) 雨水吐き口集計

流域 4 の雨水吐きにおいては、放流回数は 7 回となり、放流量は 11,325m³から 8,316m³へ約 3 割減、BOD 放流負荷量は 952kg から 451kg へ約 5 割減となった。

放流量に比べて放流負荷量の減少割合が大きいのは、降雨初期の汚濁負荷濃度の高い放流水を貯留することによる効果と考えられる。

表 5-1-1 降雨 1～12 を対象とした雨水吐き口における計算集計

	直接放流量	直接放流負荷量	貯留水量	貯留負荷量	放流回数
	m ³	kg	m ³	kg	回
通常	11,325	952	-	-	9
滞水池	8,316	451	9,234	1,559	7

5-5. 分流化

ここでは、現状合流で整備されている対策地区 B を分流化した場合の例を示す。

- (1) 分流化
- (2) 降雨と流量、汚濁負荷量の関係の例
- (3) 処理場集計
- (4) 雨水吐き口集計

(1) 分流化

分流化は、合流式下水道を污水管路と雨水管路に分離するもので、公共用水域等に未処理放流等が放流されることを防止するものであり、合流区域全域を分流化する完全分流化と区域の一部を分流化する部分分流化がある¹⁾。

また、分流化の手法として、污水管を新設し、既存の合流管を雨水管として利用する場合と、雨水管を新設し、既存の合流管を污水管として利用する場合がある（表 5-4-1）。

以下に示す例では流域 1～4 に雨水管を新設する部分分流化を想定する。

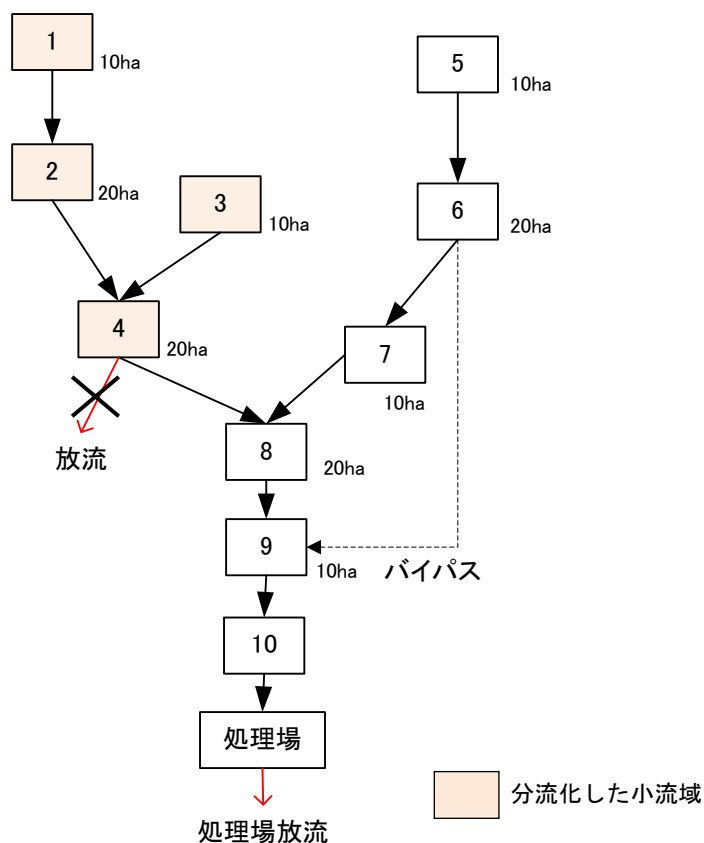


図 5-5-1 仮想モデル地区 (130ha)

表 5-5-1 分流化方式¹⁾

項目	污水管新設	雨水管新設
分流化完了後のイメージ		
新設する施設	污水：全区間新設 雨水：能力不足区間で増補管の新設	污水：新設なし 雨水：全区間新設
既設管断面の有効利用度	合流管をすべて雨水管に転用するため、有効利用度は最大。	合流管をすべて污水管に転用するため、汚水量に比べて大きな断面となる区間あり。
浸水危険度		
放流負荷量の低減	<p>管きよ(渠)整備度合 汚水管の整備とともに希釈倍率が増加。</p>	<p>管きよ(渠)整備度合 初期効果は現れにくいものの、雨水きよ(渠)の整備とともに負荷量が低減。</p>
住民への影響度	汚水ます取付け管の切替が必要となるため、切替中の住民生活に与える影響は大きい。	宅地内雨水が既設の合流ますへ接続されている場合はその切離しが必要であるため、住民への影響が出てくる。
初期投資	分流化と浸水対策を切離して行うことができ、汚水管の新設を先行させ、雨水増補管の整備を後に回した場合、初期投資を押しえられる。	浸水対策が必要な場合は、分流化の事業とともに雨水対策が同時進行となるため、トータルとしての事業費は安くなるものの初期投資が大きい。
管きよ(渠)の更新	汚水管が新設されるため、浸入水(地下水浸入)対策上有効であるが、老朽化している遮集幹線及び取付け管の更新は必要となる。	既設合流管を污水管としてそのまま使用すると、管きよ(渠)内の流速が不足し、管きよ(渠)内に土砂等が堆積し易くなるため、流速確保を含めた更新手法の検討が必要である。
総合評価	<ul style="list-style-type: none"> 宅地内の分流化の割合が高い場合は、事業当初より分流化の投資効果が現れる。 汚水管を新設するので、維持管理面に有利となる。 汚水管新設の事業となるので、住民への理解に課題が残る。 浸水被害のない区域に適している。 	<ul style="list-style-type: none"> 事業当初での分流化の投資効果は現れにくい。 合流管を污水管に転用するので、浸入水(地下水浸入)対策に課題が残る。 分流化の事業とともに雨水対策が行えるので住民の理解が得られやすい。 分流雨水から流出する汚濁が公共水域に影響を及ぼす場合には、分流雨水のノンポイント汚濁対策を講じる必要がある。

(2) 降雨と流量、汚濁負荷量の関係の例

最下流における流量および汚濁負荷量の流出解析結果の例を図 5-5-2～図 5-5-4 に示す(降雨 No1,4,7)。

分流化により、流量・汚濁負荷量とも全体的にピークが減少している。

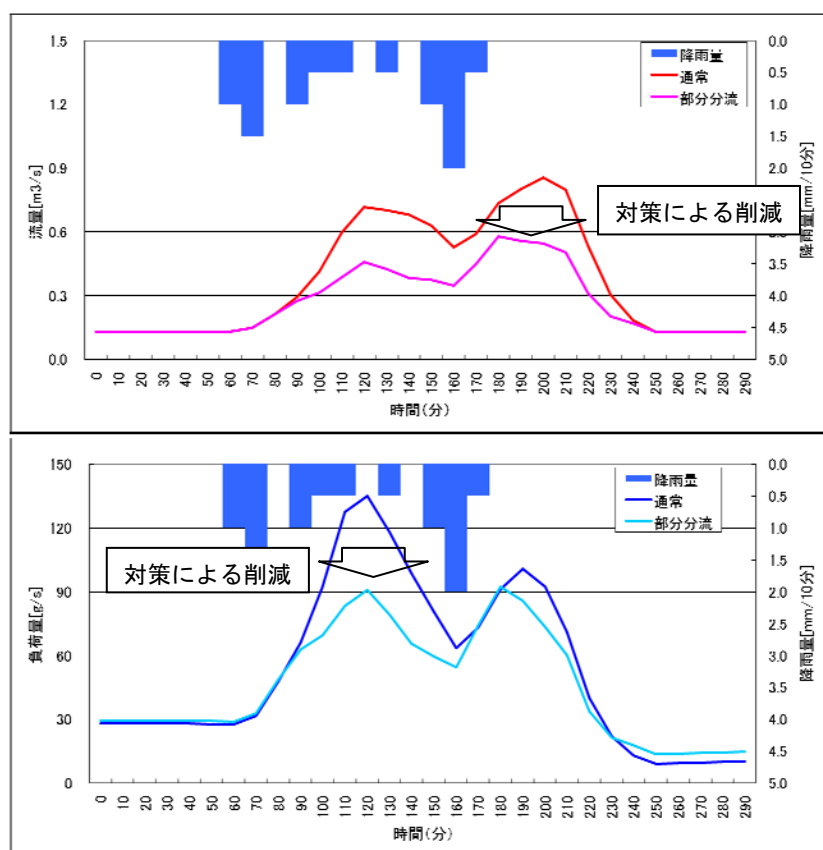


図 5-5-2 降雨 1 の解析結果 (流量、BOD)

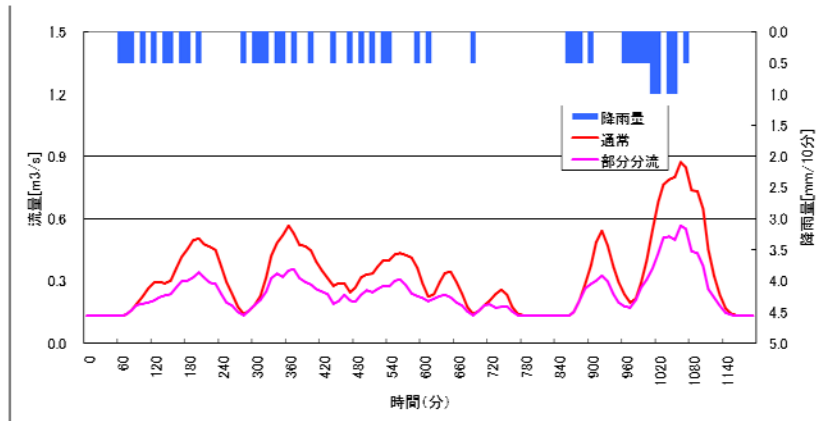


図 5-5-3 降雨 4 の解析結果 (流量、BOD)

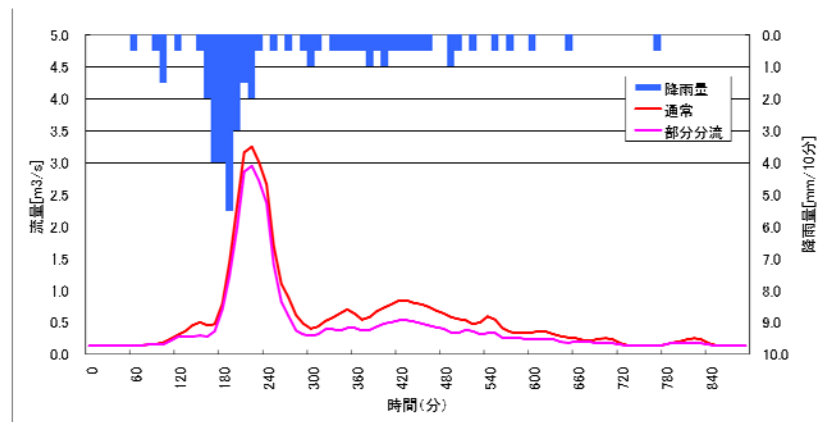


図 5-5-4 降雨 7 の解析結果 (流量、BOD)

(3) 処理場集計

降雨1、4、7を対象とした処理場の流量および汚濁負荷量の流出解析結果について、対策なしのケースとの比較を例として示す。また12降雨の集計結果の比較を表5-5-2に示す。図中、汚濁負荷量は処理後の汚濁負荷量である。

部分分流化により、処理場へ流入する水量および汚濁負荷のピークが低減している。

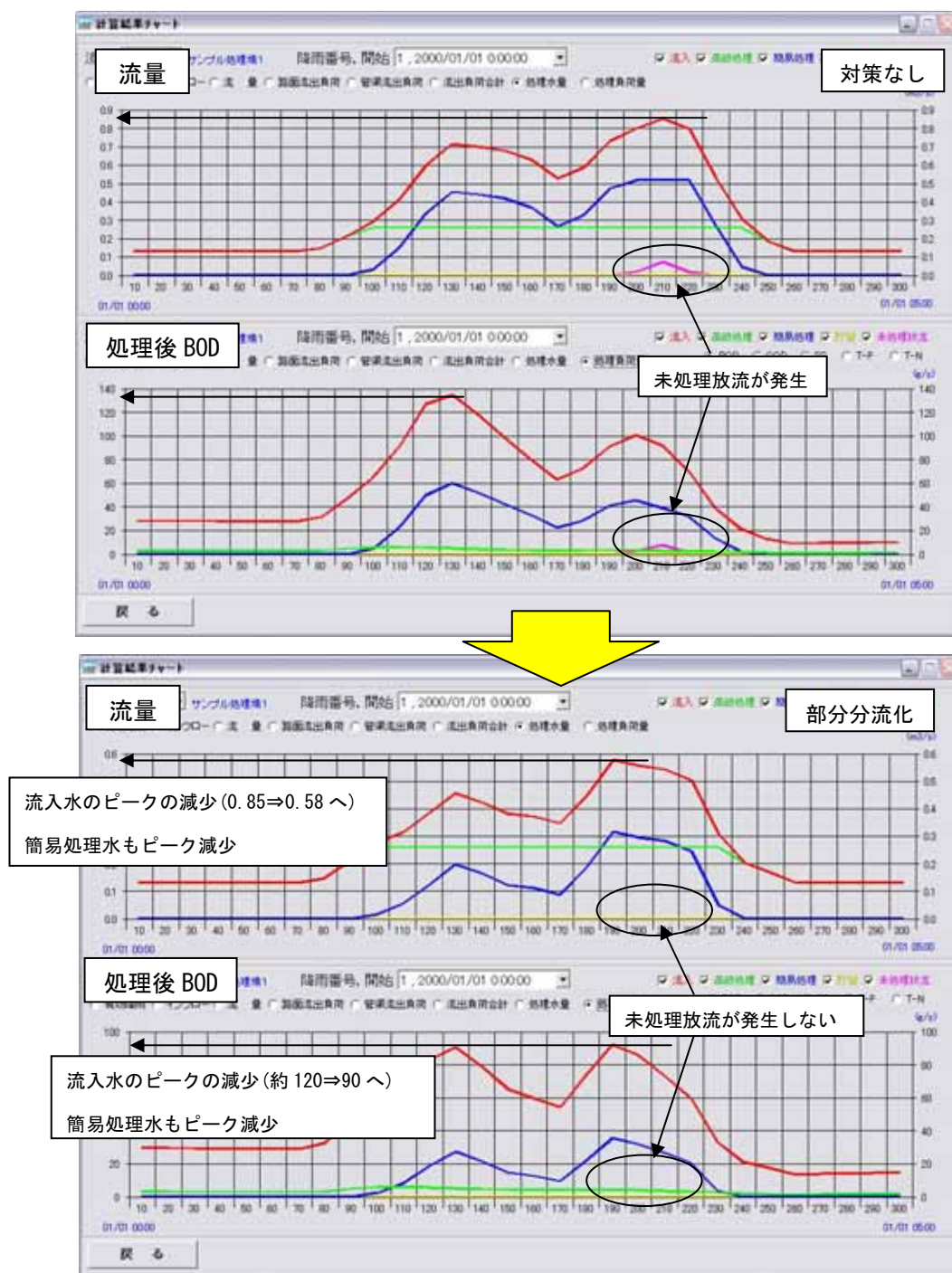


図 5-5-5 処理場計算結果 (降雨1) (上: 対策なし、下: 部分分流化)

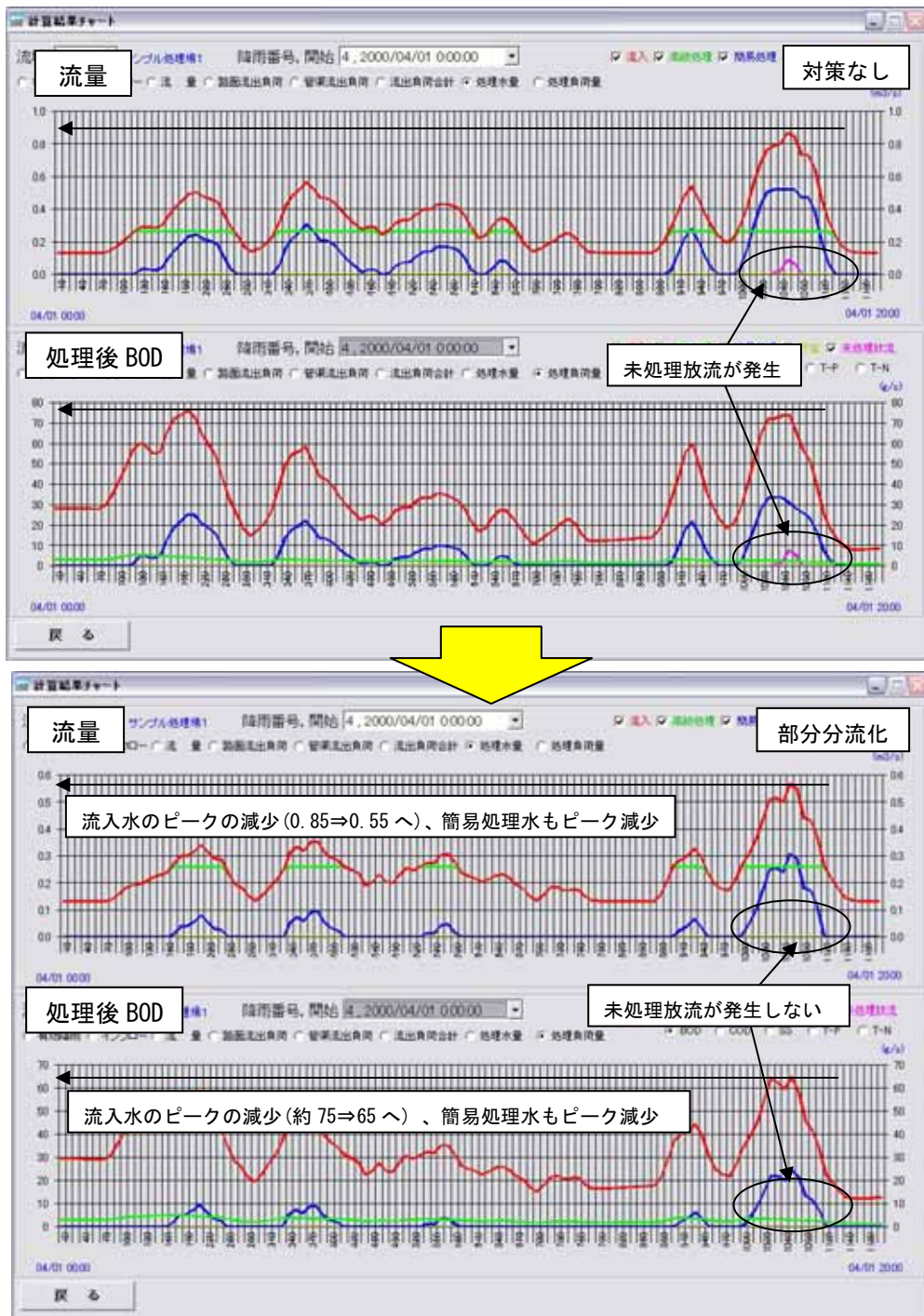


図 5-5-6 処理場計算結果 (降雨 4) (上: 対策なし、下: 部分分流化)

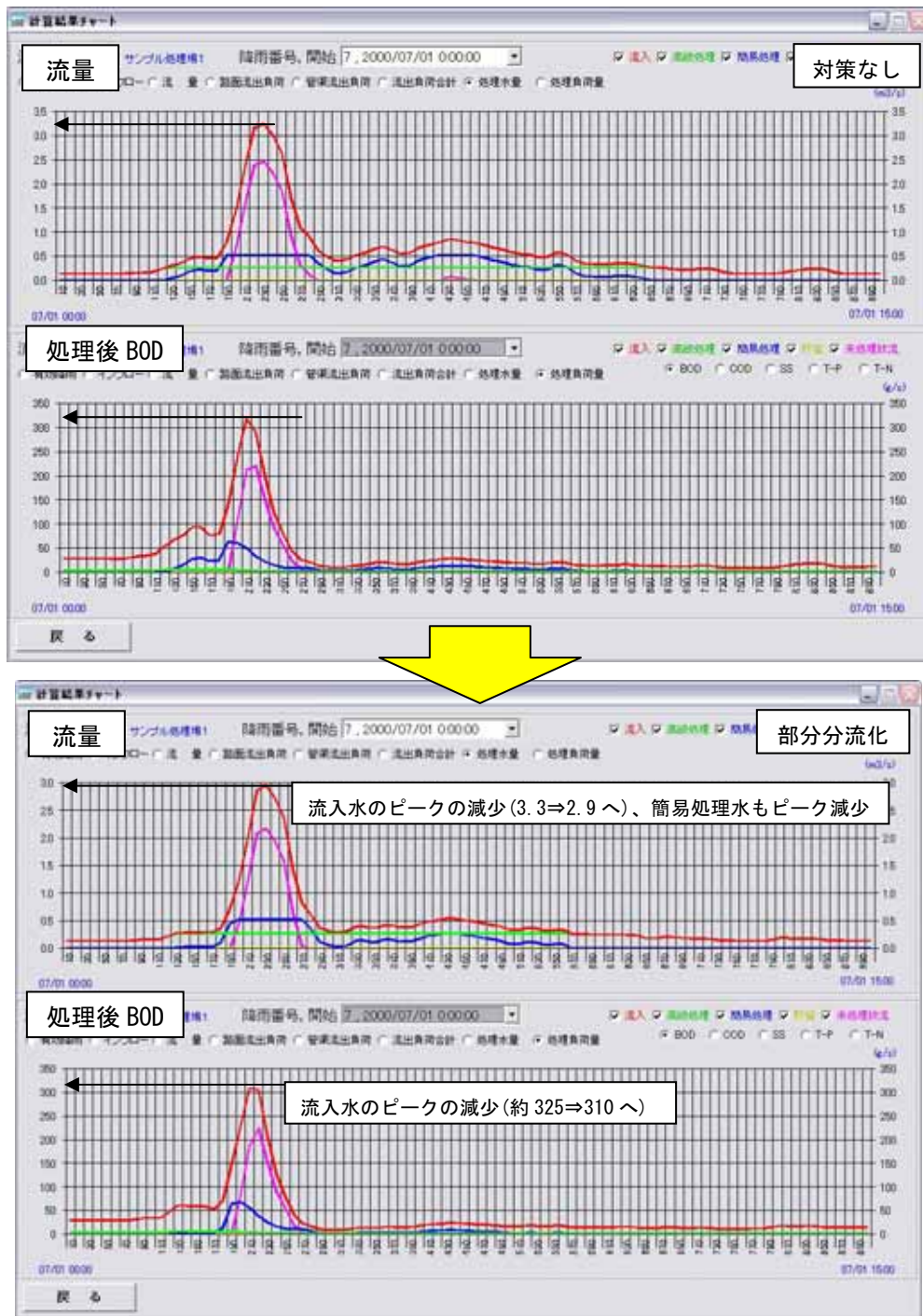


図 5-5-7 処理場計算結果 (降雨 7) (上: 対策なし、下: 部分分流化)

表 5-5-2 降雨 1～12 を対象とした処理場における計算集計

	検討ケース	高級処理	簡易処理	直接放流	計
水量(m ³)	通常	63,049	36,924	10,512	110,485
	分流化	59,911	18,185	6,282	84,377
処理前負荷量(kg)	通常	8,141	3,974	925	13,040
	分流化	8,942	2,392	519	11,853
処理後負荷量(kg)	通常	814	2,782	925	4,521
	分流化	894	1,674	519	3,087
放流回数	通常	-	-	9	
	分流化	-	-	2	

表 5-5-2 に示すとおり、部分分流化により、BOD 汚濁負荷の発生が 4,521kg から 3,087kg へ、約 32%削減された。

(4) 雨水吐き口集計

部分分流化により流域 4 の雨水吐き口は使われなくなるので、雨水吐き口からの未処理放流は放流回数、流量、負荷量とも 0 となる。

表 5-5-3 降雨 1～12 を対象とした雨水吐き口における計算集計

	直接放流量	直接放流負荷量	放流回数
	m ³	kg	回
通常	11,325	952	9
部分分流	0	0	0

5-6. 各ケースの比較

各ケースにおける水量および汚濁負荷量の集計結果の比較を以下に示す。

汚濁負荷量は、高級処理、簡易処理、貯留については処理後の負荷量を示している。

流量については、対策なしのケースと滞水池のケースは合計の雨水量が同値となるが、浸透のケースおよび部分分流のケースでは地表面から管渠へ流入する雨水量自体が減少するため合計量が小さくなっている。

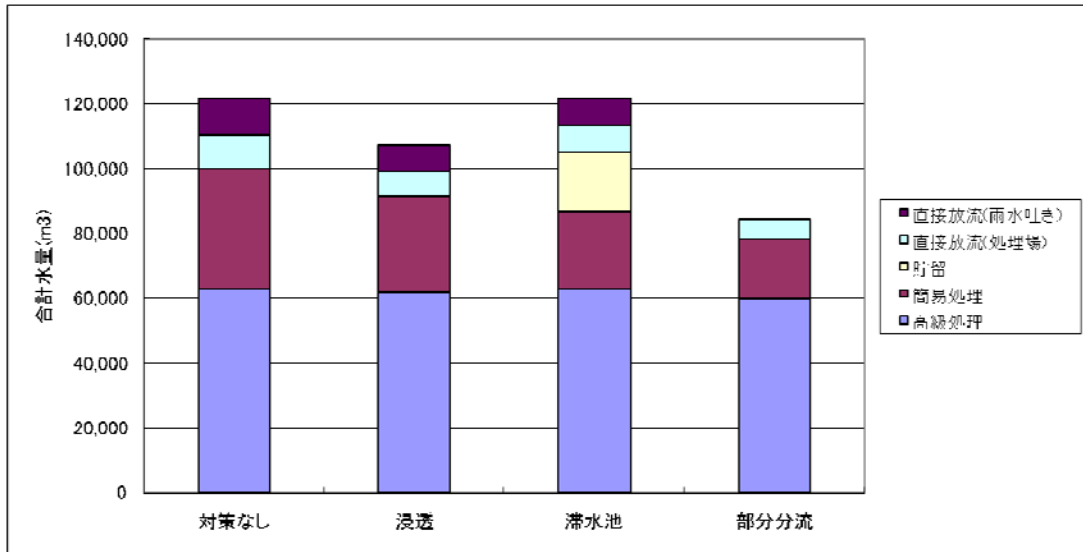


図 5-6-1 各ケースの比較 (流量)

汚濁負荷量については、今回のケースにおいては、対策なし>浸透>滞水池>部分分流の順に小さくなった。

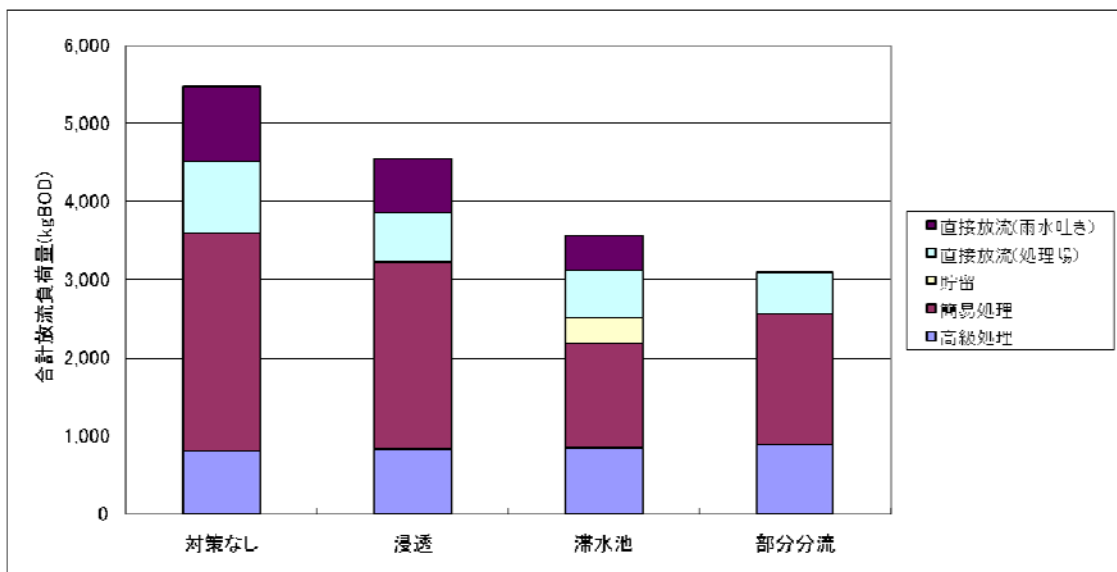


図 5-6-2 各ケースの比較 (BOD)

6. プログラム解説とプログラムリスト

6. プログラム解説とプログラムリスト

6-1. 主要変数及び各ルーチン

JAMSOOP ver.1.1.0 の解析プログラム本体は VisualBasic6.0 で構築されている。以下にプログラムで用いる主要変数と、計算過程の各ルーチンを示す。

- (1) 主要変数
- (2) 各ルーチンの概要

(1) 主要変数

表 6-1-1 共通主要変数

変数名	役割	備考
SDate	計算開始日時	
EDate	計算終了日時	
RYUSU	流域数	
maxkkou	降雨データ最大数	
TC2	降雨継続時間(通常 5 分)	

表 6-1-2 流出計算主要変数

変数名	役割	備考
area	流域面積	
Impp	不浸透面積率	
Fc	浸透域浸透能	
Ap	浸透域くぼ地割合	
Dp	浸透域くぼ地貯留高	
Ai	不浸透域くぼ地割合	
Di	不浸透域くぼ地貯留高	
RTJ	流達時間	
R	降雨データ	
TA	TA テーブル 10 分ごとの割合	
KA	TA テーブル 10 分ごとの傾き	
Saki	流下先ブロック	
Horyu	未処理放流の有無	
Bai	遮集倍率	
Bysaki	バイパス先	
dwf	流域ごとの晴天時汚水量	
dwf0	上流の流入を含む晴天時汚水量	
dwfmax	時間最大汚水量	
ByTime	バイパス流下時間	
k	SQ カーブ定数 k	
p	SQ カーブ定数 p	

表 6-1-3 汚濁負荷計算主要変数

変数名	役割	備考
Pub_RN	晴天時平均水質	
Pub_GG	限界流量	
Pub_CR	地表面残存負荷流出係数	
Pub_PSO	初期地表面負荷残存量	
Pub_RC	限界降雨強度	
Pub_a	補給係数	

表 6-1-4 算出結果変数

変数名	役割	備考
YR	有効降雨	
inf0	流域ごとのインフロー	
inf	上流分を加算したインフロー	
QAA	遮集流量	
Eflu	放流量	
ByQ	バイパス流量	
LWS3	路面負荷	
LWP3	管渠負荷	
TFUKA	流出負荷合計(遮集分)	
Eload	流出負荷合計(放流分)	
EloadB	流出負荷合計(バイパス分)	

表 6-1-5 対策施設計算結果変数

変数名	役割	備考
Chosui	貯水量	
ChoT	貯留負荷(トータル)	

(2) 各ルーチンの概要

以下に「JAMSOOP ver.1.1.0」のベースとなるプログラムの構成を示す。

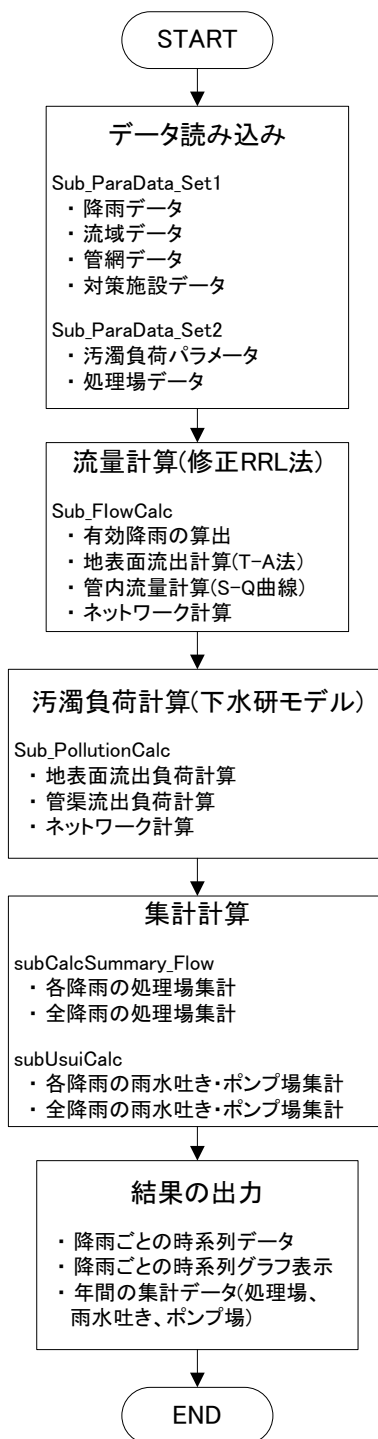


図 6-1-1 メインルーチン



図 6-1-2 サブルーチン (Sub_ParaData_Set1 : パラメータ入力 1)

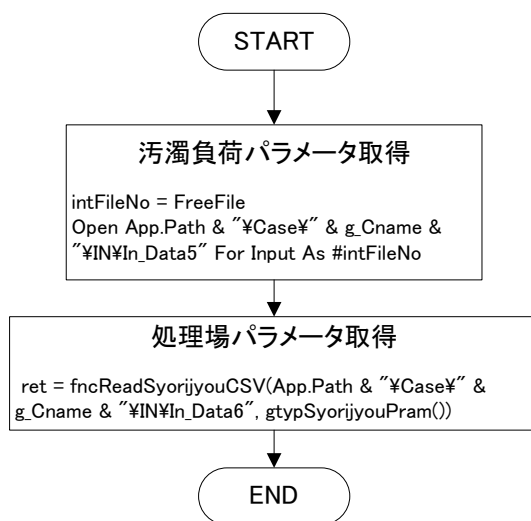


図 6-1-2 サブルーチン (Sub_ParaData_Set2 : パラメータ入力 2)

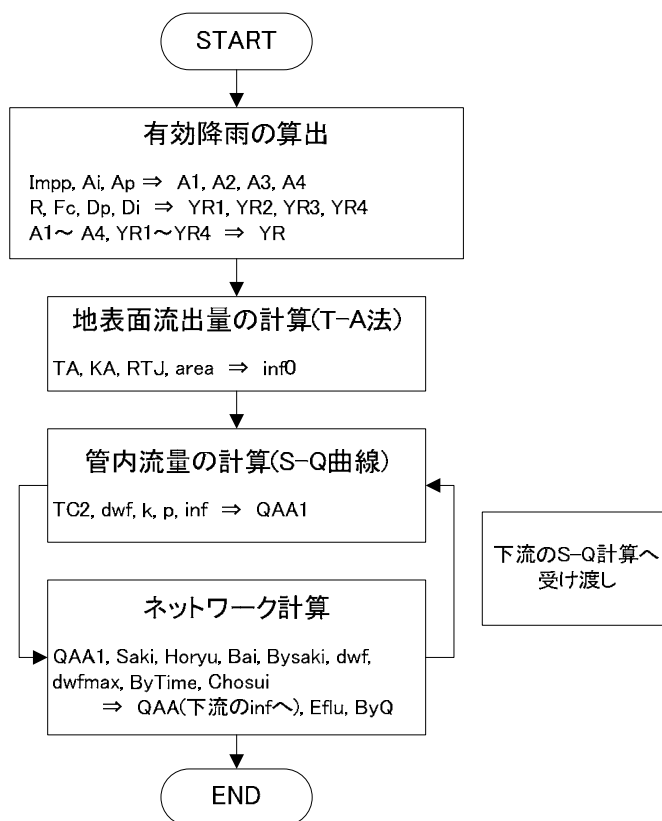


図 6-1-3 サブルーチン (Sub_Flowcalc : 流量計算)

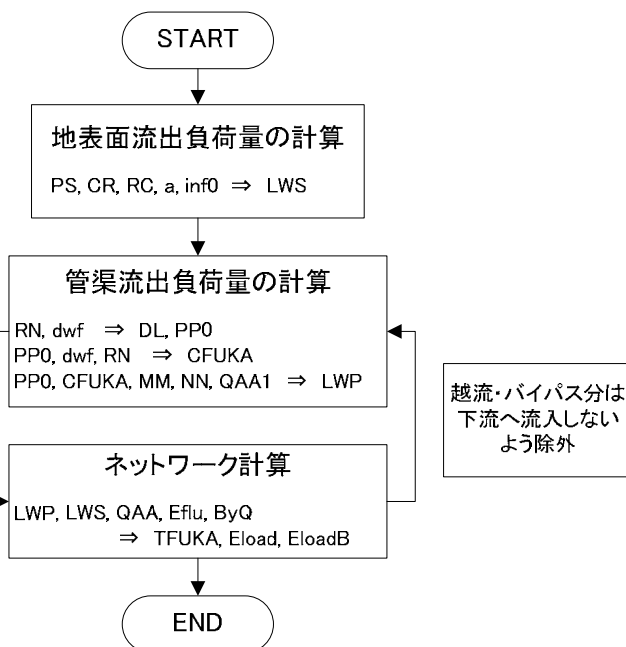


図 6-1-4 サブルーチン (Sub_Pollutioncalc : 汚濁負荷計算)

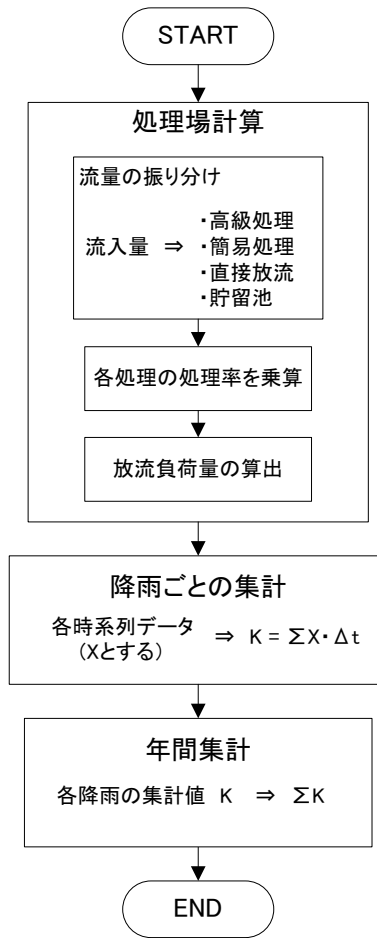


図 6-1-5 サブルーチン (subCalcSummary_Flow : 処理場集計)

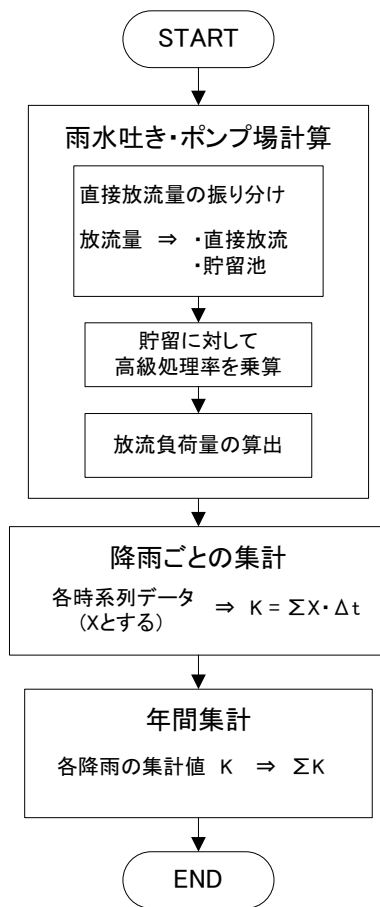


図 6-1-6 サブルーチン (subUsuiCalc : 雨水吐き・ポンプ場集計)

6-2. プログラムリスト

以下にJAMSOOP ver.1.1.0のベースとなるVisual Basicプログラムのリストを示す。JAMSOOP ver.1.1.0のプログラム全体は、インターフェースの構成や操作等に関するものも含まれているが、ここではメインの計算に関するプログラム部分のみを示す。
なおプログラム全体については、公開しているJAMSOOPのソースコード(C:\Program Files\JAMSOOP\Source_Code内)を参照されたい。

<以下計算のメインプログラム>

```
Option Explicit

*****CONSTエリア*****
Public Const CC = 1          'a=αのこと。  CC：不浸透域での負荷発生割合(通常1)
Public Const eps = 0.0002
Public Const KIRISUTE = 14  '切捨て桁数

*****共通計算用ワーク*****
Public SDate As Date        '計算開始日時
Public EDate As Date        '計算終了日時
Public RYUSU As Integer      '流域数
Public maxkkou As Integer    '降雨データ最大数
Public TC2 As Integer        '降雨継続時間(通常5分)

Public L_SyoriF As Integer   '処理区分

*****流量計算用ワーク*****
Public area0 As Single      '流域面積(ha)
Public Imppp0 As Single     '不浸透面積率(Imp)
Public Fc0 As Single        '浸透域浸透能(Fc)
Public Ap0 As Single        '浸透域凹地割合(Ap)
Public Dp0 As Single        '浸透域凹地貯留高(Dp)
Public Ai0 As Single        '不浸透域凹地割合(Ai)
Public Di0 As Single        '不浸透域凹地貯留高(Di)
Public RTJ0 As Single       '流達時間(分)

Public R0 As Single         '降雨データ
Public TA(11) As Single     'TAテーブル10分毎の割合
Public KA(11) As Single     'TAテーブル10分毎の傾き

Public Saki0 As Integer     '流下先ブロック
```

Public Horyu0	As Integer	未処理放流の有無
Public Bai0	As Single	'遮集倍率
Public Bysaki0	As Integer	'バイパス先
Public dwf0	As Single	'晴天時汚水量(m3/s)
Public dwf00	As Single	'晴天時汚水量(m3/s) (加算前)
Public dwfmax0	As Single	'時間最大汚水量
Public ByTime0	As Single	'バイパス流下時間
Public k0	As Single	'S Qカーブ定数 K
Public P0	As Single	'S Qカーブ定数 P
'雨水吐きなどの判定用		
Public Usui0	As Integer	'ネットワークパラメータの雨水吐きなどの入力値を格納
*****汚濁負荷計算用ワーク*****		
'1:BOD,2:COD,3:SS,4:T,P,5:T,N		
Public Pub_RN(5)	As Single	'晴天時平均水質(mg/L)
Public Pub_GG(5)	As Single	'限界流量(m3/s)
Public Pub_CR(5)	As Single	'地表面残存負荷流出係数(1/mm)
Public Pub_PS0(5)	As Single	'初期地表面残存負荷量(kg/ha)
Public Pub_RC(5)	As Single	'限界降雨強度 (mm/hr)
Public Pub_a(5)	As Single	'補給係数(kg/s/ha)
*****算出結果エリア*****		
Public YR0	As Single	'有効降雨
Public inf00	As Single	'インフロー (加算前)
Public inf0	As Single	'インフロー (加算後)
Public QAA0	As Single	'流量 (遮集)
Public QAA10	As Single	'流量 (遮集)
Public Eflu0	As Single	'流量 (放流量)
Public ByQ0	As Single	'流量 (バイパス量)
Public LWS30	As Single	'路面負荷(g/s)
Public LWP30	As Single	'管きよ負荷
Public TFUKA0	As Single	'流出負荷合計 (遮集)
Public Eload0	As Single	'流出負荷合計 (放流量)
Public EloadB0	As Single	'流出負荷合計 (バイパス量)
*****対策施設結果エリア*****		
Public Chosui0	As Single	'貯水量
Public ChoT0	As Single	'貯留負荷(トータル)

```

*****
' 流量計算変数データ取得処理
*****
' データ入力画面にて設定された流量計算用データを
' パブリックエリアへ設定する
*****
Public Sub Sub_ParaData_Set10
On Error GoTo Sub_ParaData_Set1_Err

Dim gyouCnt As Integer ' 行カウンタ
Dim retsuCnt As Integer ' 列カウンタ
Dim lineData As String ' 1行分のデータ
Dim strData0 As String ' 取得するデータの配列

Dim intFileNo As Integer ' ファイル番号

Dim ret As Boolean
Dim i As Integer
Dim typShisetsuPram0 As structShisetsuPram ' 対策施設パラメータの一時格納用

' ***** 保存済み降雨データ取得 *****
ret = fncReadkouCSV(App.Path & "¥Case¥" & g_CName & "¥IN¥In_Data1", gtypKouuData(), gstrStepMinute, 1)

' ***** 保存済み地表面パラメータ取得 *****
intFileNo = FreeFile
Open App.Path & "¥Case¥" & g_CName & "¥IN¥In_Data2" For Input As #intFileNo
gyouCnt = 1
retsuCnt = 0
Do While Not EOF(intFileNo)
Line Input #intFileNo, lineData

strData = Split(lineData, ",")

Select Case gyouCnt
Case 1 ' 流域数情報
RYUSU = strData(1) ' 流域数
ReDim area(RYUSU) ' 流域面積(ha)
ReDim Imppp(RYUSU) ' 不透水面積率(imp)
ReDim Fc(RYUSU) ' 浸透域浸透能(Fc)
ReDim Ap(RYUSU) ' 浸透域凹地割合(Ap)
ReDim Dp(RYUSU) ' 浸透域凹地貯留高(Dp)

```

```

ReDim Ai(RYUSU) '不浸透域凹地割合(Ai)
ReDim Di(RYUSU) '不浸透域凹地貯留高(Di)
ReDim RTJ(RYUSU) '流達時間(分)

Else
MsgBox "流域数データが不正です。" + Chr(13) + Chr(13) + "データ再度登録してください。", vbCritical, "データ構成不正"
Close #intFileNo
Exit Sub
End If
Case 2, 3
'ヘッダ情報は使用しない
Case Else
area(gyouCnt - 3) = strData(1) '流域面積(ha)
Impp(gyouCnt - 3) = strData(2) '不浸透面積率(Imp)
Fc(gyouCnt - 3) = strData(3) '浸透域浸透能(Fc)
Ap(gyouCnt - 3) = strData(4) '浸透域凹地割合(Ap)
Dp(gyouCnt - 3) = strData(5) '浸透域凹地貯留高(Dp)
Ai(gyouCnt - 3) = strData(6) '不浸透域凹地割合(Ai)
Di(gyouCnt - 3) = strData(7) '不浸透域凹地貯留高(Di)
RTJ(gyouCnt - 3) = strData(8) '流達時間(分)

End Select

'行数をカウントアップ
gyouCnt = gyouCnt + 1

Loop

Close #intFileNo

'*****等到達時間パラメータ取得*****
intFileNo = FreeFile
Open App.Path & "¥Case¥" & g_Cname & "¥IN¥In_Data3" For Input As #intFileNo
gyouCnt = 1
retsuCnt = 0
Do While Not EOF(intFileNo)
Line Input #intFileNo, lineData

strData = Split(lineData, ",")

If gyouCnt = 2 Then
For retsuCnt = 0 To UBound(strData)
TA(retsuCnt) = strData(retsuCnt) 'T A テーブル10分毎の割合
Next retsuCnt
End If

```

```

行数をカウントアップ
gyouCnt = gyouCnt + 1
Loop

Close #intFileNo

'*****ネットワークパラメータ取得*****
intFileNo = FreeFile
Open App.Path & "\Case¥" & g_Cname & "\IN¥In_Data4" For Input As #intFileNo

ReDim Saki(RYUSU) '流下先ブロック
ReDim Horyu(RYUSU) '未処理放流の有無
ReDim Bai(RYUSU) '遮集倍率
ReDim Bysaki(RYUSU) 'バイパス先
ReDim dwf(RYUSU) '晴天時汚水量(m3/s)
ReDim dwf0(RYUSU) '晴天時汚水量(m3/s) (加算前)
ReDim dwfmax(RYUSU) '時間最大汚水量
ReDim ByTime(RYUSU) 'バイパス流下時間
ReDim k(RYUSU) 'SQカーブ定数 K
ReDim P(RYUSU) 'SQカーブ定数 P
ReDim Usui(RYUSU) '雨水吐きなど

gyouCnt = 1
retsuCnt = 0
Do While Not EOF(intFileNo)
Line Input #intFileNo, lineData

strData = Split(lineData, ";")

Select Case gyouCnt
Case 1, 2
'ヘッダ情報は使用しない
Case Else
Saki(gyouCnt - 2) = strData(1) '流下先ブロック
Usui(gyouCnt - 2) = strData(2) '雨水吐きなど:0:なし、1:雨水吐き、2:ポンプ場、3:バイパス
'雨水吐きとポンプ場の場合は放流の変数に1を設定、バイパスの場合は2を設定
Select Case Usui(gyouCnt - 2)
Case 0
Horyu(gyouCnt - 2) = 0 '未処理放流の有無(放流なし)
Case 1, 2

```

```

        Horyu(gyouCnt - 2) = 1      '未処理放流の有無(放流あり)
    Case 3
        Horyu(gyouCnt - 2) = 2      '未処理放流の有無(バイパス)
    End Select

    Bai(gyouCnt - 2) = strData(3)    '遮集倍率
    Bysaki(gyouCnt - 2) = strData(4)  'バイパス先
    dwf(gyouCnt - 2) = strData(5)    '晴天時汚水量(m3/s)
    dwf0(gyouCnt - 2) = strData(5)   '晴天時汚水量(m3/s) (加算前保存)
    dwfmax(gyouCnt - 2) = strData(6) '時間最大汚水量
    ByTime(gyouCnt - 2) = strData(7) 'バイパス流下時間
    k(gyouCnt - 2) = strData(8)     'SQカーブ定数K
    P(gyouCnt - 2) = strData(9)     'SQカーブ定数P

    End Select

    行数をカウントアップ
    gyouCnt = gyouCnt + 1
Loop

'***** 対策施設パラメータ取得*****
ret = fncReadShisetsuCSV(App.Path & "¥Case¥" & g_Cname & "¥IN¥In_Data7", typShisetsuPram0)

'計算時のループ処理回避のため対策施設パラメータは配列を拡張する(メモリ使用量は少し増えますが、ほとんどが長さ0の文字列なので無視できる量です)
ReDim gtypShisetsuPram(RYUSU)
For i = 1 To UBound(typShisetsuPram)
    gtypShisetsuPram(i) = typShisetsuPram(i)
Next i

Sub_ParaData_Set1_Exit:
    Close #intFileNo
Exit Sub
Sub_ParaData_Set1_Err:
    MsgBox Err.Description
Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation]:" & "Sub_ParaData_Set1", "流量計算データ取得処理にてエラーが発生しました。")
Resume Sub_ParaData_Set1_Exit

End Sub

'*****
' 汚濁負荷計算変数データ取得処理
'*****

```

```

' データ入力画面にて設定された汚濁負荷計算用データを
' パブリックエリアへ設定する
*****
Public Sub Sub_ParaData_Set20
'On Error GoTo Sub_ParaData_Set2_Err
On Error GoTo 0

Dim gyokuCnt As Integer ' 行カウンタ
Dim retsuCnt As Integer ' 列カウンタ
Dim lineData As String ' 1行分のデータ
Dim strData0 As String ' 取得するデータの配列

Dim intFileNo As Integer ' ファイル番号

' *****保存済み汚濁負荷データ取得*****
intFileNo = FreeFile
Open App.Path & "\Case¥" & g_GName & "\IN¥In_Data5" For Input As #intFileNo
gyokuCnt = 1
retsuCnt = 0
Do While Not EOF(intFileNo)

Line Input #intFileNo, lineData
strData = Split(lineData, ",")

Select Case gyokuCnt
Case 1 'ヘッダ行
Case 2 '晴天時平均水質
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_RN(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_RN(retsuCnt) = 0
End If
Next retsuCnt
Case 3 '限界流量
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_GG(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_GG(retsuCnt) = 0

```

```

End If
Next retsuCnt
Case 4 '空白行
Case 5 '地表面残存負荷流出係数
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_CR(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_CR(retsuCnt) = 0
End If
Next retsuCnt
Case 6 '初期地表面残存負荷量
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_PS0(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_PS0(retsuCnt) = 0
End If
Next retsuCnt
Case 7 '眼界降雨強度
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_RC(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_RC(retsuCnt) = 0
End If
Next retsuCnt
Case 8 '補給係数
For retsuCnt = 1 To 5
If IsNumeric(strData(retsuCnt + 1)) Then
Pub_a(retsuCnt) = strData(retsuCnt + 1)
Else
Pub_a(retsuCnt) = 0
End If
Next retsuCnt
End Select
'行数をカレントアップ
gyouCnt = gyokuCnt + 1
Loop
Dim ret As Boolean

```



```

'*****処理場データ取得*****
ret = fncReadSyorijyouCSV(App.Path & "¥IN¥In_Data6", g_CName & "¥IN¥In_Data6", gtypSyorijyouPram())

Sub_ParaData_Set2_Exit:
  Close #intFileNo
  Exit Sub
Sub_ParaData_Set2_Err:
  MsgBox Err.Description
  Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation]:" & "Sub_ParaData_Set2", "流量計算データ取得処理にてエラーが発生しました。")
  Resume Sub_ParaData_Set2_Exit

End Sub

*****
' 流量計算処理
*****
' パブリックエリアの変数データを元に流量計算を行う
*****

Public Sub Sub_FrowCalc()
  On Error GoTo Sub_FrowCalc_Err
  Dim i As Integer
  Dim j As Integer
  Dim ij As Integer
  Dim n As Integer
  Dim NNN As Integer
  Dim MMM As Integer

  *****不浸透域の流出計算*****
  'n=流域数 i = 雨量 mm
  Dim RS1 As Single
  Dim RS2 As Single
  Dim TC3 As Single
  Dim YR1 As Single
  Dim YR2 As Single
  Dim YR3 As Single
  Dim YR4 As Single

  *****ニュートンを法を使用*****
  Dim Time2 As Integer

```

```

Dim Time3 As Integer
Dim X As Single
Dim a As Single
Dim B As Single
Dim s0 As Single
Dim Flow2 As Single
Dim Flow3 As Single
Dim TTC2 As Single

'TC2 / 60

Dim intLoop As Integer 'ループ用カウンタ
Dim intCount As Integer '降雨毎の最大計算ステップをカウントするためのワーク

Dim intStepSecond As Integer '1 ステップあたりの秒数
Dim sngDwfMax As Single '対策施設の計算で使用する流量

'ループに対応するため、格納用の変数の範囲を最初に宣言
maxkkou = UBound(gtypKouuData(1).strKouuData) '最大計算ステップ数は降雨データを格納している変数の配列数
TTC2 = gstrStepMinute '降雨時間(分) / ステップ

intStepSecond = CInt(gstrStepMinute) * 60 '1 ステップあたりの秒数を設定

*****有効降雨計算変数の定義*****
ReDim YR(UBound(gtypKouuData), RYUSU, maxkkou) As Single

*****流量の重ね合わせ*****
ReDim inf0(UBound(gtypKouuData), RYUSU, maxkkou) As Single
ReDim inf1(UBound(gtypKouuData), RYUSU, maxkkou) As Single

***SQ 流出量計算変数の定義***
ReDim Eflu(UBound(gtypKouuData), RYUSU, maxkkou) As Single
ReDim ByQ(UBound(gtypKouuData), RYUSU, maxkkou) As Single
ReDim QAA(UBound(gtypKouuData), RYUSU, maxkkou) As Single
ReDim QAA1(UBound(gtypKouuData), RYUSU, maxkkou) As Single

'対策施設の貯留量
ReDim Chosui(UBound(gtypKouuData), RYUSU, maxkkou) As Single '対策施設の貯留量

*****
'複数の降雨データでループ化
MenuF.lblCalcStatus.Visible = True
For intLoop = 1 To UBound(gtypKouuData)

```

```
MenuFtblCalcStatus.Caption = "流量計算実行中です" & vbCrLf & "(" & intLoop & "/" & intLoop & ")" & UBound(gtYPKouuData) & "件目"
DoEvents
```

```
'ローカル変数をループごとに初期化
```

```
*****不浸透域の流出計算*****
```

```
RS1 = 0
```

```
RS2 = 0
```

```
TC3 = 0
```

```
YR1 = 0
```

```
YR2 = 0
```

```
YR3 = 0
```

```
YR4 = 0
```

```
Time2 = 0
```

```
Time3 = 0
```

```
X = 0
```

```
a = 0
```

```
B = 0
```

```
s0 = 0
```

```
Flow2 = 0
```

```
Flow3 = 0
```

```
TTC2 = 0
```

```
'読み込んだ行ごとのステップ数
```

```
maxkkou = gtYPKouuData(intLoop).lngStepCount
```

```
'計算用のワーク変数を初期化
```

```
*****面積変換*****
```

```
ReDim A1(RYUSU)
```

```
'不浸透域で凹地貯留がない面積
```

```
ReDim A2(RYUSU)
```

```
'不浸透域で凹地貯留がある面積
```

```
ReDim A3(RYUSU)
```

```
'浸透域で凹地貯留がない面積
```

```
ReDim A4(RYUSU)
```

```
'浸透域で凹地貯留がある面積
```

```
*****有効降雨計算変数の定義*****
```

```
ReDim YRR(RYUSU, maxkkou) As Single
```

```
ReDim FCC(RYUSU)
```

```
ReDim SA(RYUSU, maxkkou)
```

```
*****流達時間に対する降雨継続時間(分)ごとの割合計算*****
```

```
ReDim Wari(RYUSU, 15) As Single
```

```
ReDim kkk(RYUSU) As Integer
```

```
'流達時間の分割数
```

```

ReDim MRIT(RYUSU, 15) As Single
ReDim MRIT2(RYUSU, 15) As Single

*****降雨継続時間ごとの面積と雨量計算*****
ReDim AA(RYUSU, 15) As Single
ReDim aaa(RYUSU, 15) As Single
ReDim MRT(RYUSU, 11) As Single
ReDim RAIN(RYUSU, maxkkou + 30, 20) As Single 'n 番目流域、m 番目降雨 (降雨 index は未来方向に 2 時間程度必要)、到達時間分割 No

***SQ 流出量計算変数の定義*****
ReDim s(maxkkou) As Single
ReDim Q(RYUSU, maxkkou) As Single
ReDim QAU(RYUSU, maxkkou) As Single '雨水量

晴天時汚水量(m3/s)の初期化
ReDim dwf(UBound(dwf0))
dwf = dwf0

'対策施設の貯留容量
ReDim QAA2(RYUSU, maxkkou) As Single '貯留し切れなかった分

*****面積変換*****
'流域面積算出
For n = 1 To RYUSU
    '不透透面積率にて判定
    If Impp(n) = 0 Then
        A1(n) = 0
        A2(n) = 0
        A3(n) = 0
        A4(n) = 0
    Else
        A1(n) = Impp(n) * (1 - Ai(n) / 100)
        A2(n) = Impp(n) * Ai(n) / 100
        A3(n) = (1 - Impp(n)) * (1 - Ap(n) / 100)
        A4(n) = (1 - Impp(n)) * Ap(n) / 100
    End If
Next n

*****有効降雨計算変数の定義*****
RS1 = 0
RS2 = 0
TC3 = 60 / TC2

```

```

For n = 1 To RYUSU
    For i = 1 To maxkkou
        流域面積にて判定
        If area(n) = 0 Then
            YR(intLoop, n, i) = 0
            GoTo 400
        End If
        YR1 = R(i)
        RS1 = RS1 + R(i)
        YR1 = gtypKouuData(intLoop).strKouuData(i)
        RS1 = RS1 + gtypKouuData(intLoop).strKouuData(i)
        If RS1 <= Di(n) Then
            YR2 = 0
        Else
            If RS1 > Di(n) Then
                YR2 = RS1 - RS2 - Di(n)
            Else
                YR2 = 0
            End If
        End If
        RS2 = RS2 + YR2
    *****浸透域の流出計算*****
    SA(n, 0) = 0
    FCC(n) = Round(Fc(n) / TC3, 5)
    If R(i) <= FCC(n) Then
        YR3 = 0
    Else
        YR3 = R(i) - FCC(n)
    End If
    If gtypKouuData(intLoop).strKouuData(i) <= FCC(n) Then
        YR3 = 0
    Else
        YR3 = gtypKouuData(intLoop).strKouuData(i) - FCC(n)
    End If

```

100

```

'200      SA(n, i) = SA(n, i - 1) + R(i) - FCC(n)
200      SA(n, i) = SA(n, i - 1) + gtypKouuData(intLoop).strKouuData(i) - FCC(n)

      YR4 = 0
      If SA(n, i) <= 0 Then
        SA(n, i) = 0
        GoTo 300
      End If

      If SA(n, i) >= 0 And SA(n, i) <= Dp(n) Then
        GoTo 300
      End If

      If SA(n, i) > Dp(n) Then
        YR4 = SA(n, i) - Dp(n)
        SA(n, i) = Dp(n)
      End If

      *****有効降雨の計算*****
      YRR(n, i) = A1(n) * YR1 + A2(n) * YR2 + A3(n) * YR3 + A4(n) * YR4
      YR(intLoop, n, i) = Round(YRR(n, i) + 0.0001, 2)

      Next i
      RS1 = 0
      RS2 = 0
      DoEvents
      Next n
      Erase SA

      *****降雨継続時間ごとのTA計算*****
      *****各ステップの傾き計算*****
      For i = 1 To 10
        KA(i) = (TA(i) - TA(i - 1)) / 10
      Next i

      *****流達時間に対する降雨継続時間(分)ごとの割合計算*****
      For n = 1 To RYUSU
        '流達時間(分)にて判定
        If RTJ(n) = 0 Then
          Wari(n, 1) = 1
        Else

```

300

400

```

For i = 1 To 13
  Wari(n, i) = TC2 * i / RTJ(n)
  If Wari(n, i) >= 1 Then
    Wari(n, i) = 1
    kkk(n) = i
  Exit For
End If
Next i
End If

*****降雨継続時間ごとの面積率計算*****
For i = 1 To kkk(n)
  If Wari(n, i) <= 0.1 Then
    MRIT(n, i) = TA(0) / 100 + KA(1) * (Wari(n, i) - 0)
  ElseIf 0.1 < Wari(n, i) And Wari(n, i) <= 0.2 Then
    MRIT(n, i) = TA(1) / 100 + KA(2) * (Wari(n, i) - 0.1)
  ElseIf 0.2 < Wari(n, i) And Wari(n, i) <= 0.3 Then
    MRIT(n, i) = TA(2) / 100 + KA(3) * (Wari(n, i) - 0.2)
  ElseIf 0.3 < Wari(n, i) And Wari(n, i) <= 0.4 Then
    MRIT(n, i) = TA(3) / 100 + KA(4) * (Wari(n, i) - 0.3)
  ElseIf 0.4 < Wari(n, i) And Wari(n, i) <= 0.5 Then
    MRIT(n, i) = TA(4) / 100 + KA(5) * (Wari(n, i) - 0.4)
  ElseIf 0.5 < Wari(n, i) And Wari(n, i) <= 0.6 Then
    MRIT(n, i) = TA(5) / 100 + KA(6) * (Wari(n, i) - 0.5)
  ElseIf 0.6 < Wari(n, i) And Wari(n, i) <= 0.7 Then
    MRIT(n, i) = TA(6) / 100 + KA(7) * (Wari(n, i) - 0.6)
  ElseIf 0.7 < Wari(n, i) And Wari(n, i) <= 0.8 Then
    MRIT(n, i) = TA(7) / 100 + KA(8) * (Wari(n, i) - 0.7)
  ElseIf 0.8 < Wari(n, i) And Wari(n, i) <= 0.9 Then
    MRIT(n, i) = TA(8) / 100 + KA(9) * (Wari(n, i) - 0.8)
  ElseIf 0.9 < Wari(n, i) And Wari(n, i) <= 1 Then
    MRIT(n, i) = TA(9) / 100 + KA(10) * (Wari(n, i) - 0.9)

```

```

End If
MRT2(n, i) = Int(10000 * (MRT(n, i) + 0.000009)) / 10000
Next i
DoEvents
Next n

*****降雨継続時間ごとの面積と雨量計算*****
For n = 1 To RYUSU
DoEvents
*****流域面積の読み*****
AA(n, 0) = 0
For jj = 1 To kkk(n)
aaa(n, jj) = MRT2(n, jj) * area(n) / 100 'T-A の各ステップにおける累加面積 ha→km2
AA(n, jj) = aaa(n, jj) - aaa(n, jj - 1) 'T-A の各ステップにおける各面積
Next jj

*****降雨継続時間ごとの面積率計算*****
MRT(n, 0) = 0
For jj = 1 To kkk(n)
MRT(n, jj) = MRT2(n, jj) - MRT2(n, jj - 1)
Next jj

*****各面積の流量・負荷量計算(面積率で計算)*****
YR(intLoop, n, 0) = 0
NNN = 1
For jj = 1 To kkk(n)
For i = 0 To maxkkou - 1
RAIN(n, i + NNN, jj) = MRT(n, jj) * YR(intLoop, n, i + 1) * TC3
Next i
NNN = NNN + 1
Next jj
DoEvents
Next n

*****流量の重ね合わせ*****
For n = 1 To RYUSU
inf(intLoop, n, 1) = 0

```



```

For i = 1 To maxkkou
  For jj = 1 To kkk(n)
    inf(intLoop, n, i) = inf(intLoop, n, i) + RAIN(n, i, jj) 'インフロー計算
  Next jj
  info(intLoop, n, i) = inf(intLoop, n, i) '加算前インフロー保存
Next i
Next n

*****SQ カークによる流出量計算*****
****SQ 流出量計算変数の定義****
TTC2 = TC2 / 60

For n = 1 To RYUSU
  Time2 = Round((RTJ(n) + 0.00001) / TTC2, 0)
  dwf(Saki(n)) = dwf(Saki(n)) + dwf(n)
  X = dwf(n)
  If area(n) = 0 Then
    s0 = k(n) * (dwf(n) / 10 * 360) ^ P(n)
    a = Sgn(X) * k(n) * Abs(X) ^ P(n) - TTC2 / 2 * (inf(intLoop, n, 1) + dwf(n) / 10 * 360) + TTC2 / 2 * X - s0
  Else
    s0 = k(n) * (dwf(n) / area(n) * 360) ^ P(n)
    a = Sgn(X) * k(n) * Abs(X) ^ P(n) - TTC2 / 2 * (inf(intLoop, n, 1) + dwf(n) / area(n) * 360) + TTC2 / 2 * X - s0
  End If
  '0のマイナスべき乗が不可なので対応を行う
  If X = 0 Then
    B = 99999999
  Else
    B = Sgn(X) * P(n) * k(n) * Abs(X) ^ (P(n) - 1) + TTC2 / 2
  End If
  If Abs(a) < eps Then
    GoTo 450
  Else
    X = X - a / B
    GoTo 430
  End If

```

450

```

Q(n, 1) = X
If area(n) = 0 Then
  s(1) = TTC2 / 2 * (inf(intLoop, n, 1) + dwf(n) / 10 * 360) - TTC2 / 2 * Q(n, 1) + s0
Else
  s(1) = TTC2 / 2 * (inf(intLoop, n, 1) + dwf(n) / area(n) * 360) - TTC2 / 2 * Q(n, 1) + s0
End If

```

'455

```

If area(n) = 0 Then
  QAA1(intLoop, n, 1) = 10 * Q(n, 1) / 360
Else
  QAA1(intLoop, n, 1) = area(n) * Q(n, 1) / 360
End If

```

'対策施設の計算処理

'対策施設のデータが変数に格納されていたら、対策施設の計算を行う

If IsNumeric(gtypShisetsuPram(n).strNo) = True Then

'最初の降雨で貯留される量を計算

```

If QAA1(intLoop, n, 1) > CSng(gtypShisetsuPram(n).strChoryuStart) Then
  Chosui(intLoop, n, 1) = (QAA1(intLoop, n, 1) - CSng(gtypShisetsuPram(n).strChoryuStart)) * intStepSecond
  QAA(intLoop, n, 1) = CSng(gtypShisetsuPram(n).strChoryuStart)
  QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)

```

GoTo 460

Else

Chosui(intLoop, n, 1) = 0

End If

End If

457

```

Select Case Horyu(n)
Case 1
  If QAA1(intLoop, n, 1) > Bai(n) * dwfmax(n) Then
    Eflu(intLoop, n, 1) = QAA1(intLoop, n, 1) - Bai(n) * dwfmax(n)
  QAA(intLoop, n, 1) = Bai(n) * dwfmax(n)
  QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)
Else
  Eflu(intLoop, n, 1) = 0

```

```

QAA(intLoop, n, 1) = QAA1(intLoop, n, 1)
QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)
End If
Case 2
If QAA1(intLoop, n, 1) > Bai(n) * dwfmax(n) Then
Eflu(intLoop, n, 1) = 0
ByQ(intLoop, n, 1) = QAA1(intLoop, n, 1) - Bai(n) * dwfmax(n)
QAA(intLoop, n, 1) = Bai(n) * dwfmax(n)
QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)
Else
Eflu(intLoop, n, 1) = 0
ByQ(intLoop, n, 1) = 0
QAA(intLoop, n, 1) = QAA1(intLoop, n, 1)
QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)
End If
Case Else
Eflu(intLoop, n, 1) = 0
QAA(intLoop, n, 1) = QAA1(intLoop, n, 1)
QAU(n, 1) = QAA(intLoop, n, 1) - dwf(n)
End Select
***流下***
If -Time2 <= 0 Then
Flow2 = 0
Else
Flow2 = QAU(n, 1 - Time2)
End If
If Saki(n) = 0 Then
Else
If area(Saki(n)) = 0 Then
inf(intLoop, Saki(n), 1) = inf(intLoop, Saki(n), 1) + (Flow2 / 10 * 360)
Else
inf(intLoop, Saki(n), 1) = inf(intLoop, Saki(n), 1) + (Flow2 / area(Saki(n)) * 360)
End If
End If
***バババズ***
If Horyu(n) = 2 Then
Time3 = Round(ByTime(n) / 5, 0)
If -Time3 <= 0 Then

```

```

Flow3 = 0
Else
  Flow3 = ByQ(intLoop, n, 1 - Time3)
End If

If area(Bysaki(n)) = 0 Then
  inf(intLoop, Bysaki(n), 1) = inf(intLoop, Bysaki(n), 1) + (Flow3 / 10 * 360)
Else
  inf(intLoop, Bysaki(n), 1) = inf(intLoop, Bysaki(n), 1) + (Flow3 / area(Bysaki(n)) * 360)
End If

X = Q(n, 1)

For i = 2 To maxkkou
  DoEvents
  If area(n) = 0 Then
    a = Sgn(X) * k(n) * Abs(X) ^ P(n) - TTC2 / 2 * (inf(intLoop, n, i - 1) + inf(intLoop, n, i) + 2 * dwf(n) / 10 * 360) + TTC2 / 2 * (Q(n, i - 1) + X) - s(i - 1)
  Else
    a = Sgn(X) * k(n) * Abs(X) ^ P(n) - TTC2 / 2 * (inf(intLoop, n, i - 1) + inf(intLoop, n, i) + 2 * dwf(n) / area(n) * 360) + TTC2 / 2 * (Q(n, i - 1) + X) - s(i - 1)
  End If
  '0のマイナスべき乗が不可なので対応を行う
  If X = 0 Then
    B = 99999999
  Else
    B = Sgn(X) * P(n) * k(n) * Abs(X) ^ (P(n) - 1) + TTC2 / 2
  End If

  If Abs(a) < eps Then
    GoTo 600
  Else
    X = X - a / B
  End If

  If area(n) = 0 Then
    If X < dwf(n) / 10 * 360 Then
      X = dwf(n) / 10 * 360
      GoTo 600
    End If
  Else
    If X < dwf(n) / area(n) * 360 Then

```

500

```

        X = dwf(n) / area(n) * 360
        GoTo 600
    End If
End If
GoTo 500
End If

Q(n, i) = X

If area(n) = 0 Then
    s(i) = TTC2 / 2 * (inf(intLoop, n, i - 1) + inf(intLoop, n, i) + 2 * dwf(n) / 10 * 360) - TTC2 / 2 * (Q(n, i - 1) + Q(n, i)) + s(i - 1)
Else
    s(i) = TTC2 / 2 * (inf(intLoop, n, i - 1) + inf(intLoop, n, i) + 2 * dwf(n) / area(n) * 360) - TTC2 / 2 * (Q(n, i - 1) + Q(n, i)) + s(i - 1)
End If

If area(n) = 0 Then
    QAA1(intLoop, n, i) = 10 * Q(n, i) / 360
Else
    QAA1(intLoop, n, i) = area(n) * Q(n, i) / 360
End If

'対策施設の計算処理
If IsNumeric(gttypShisetsuPram(n).strNo) = True Then
    GoTo 653
Else
    GoTo 655
End If

'貯留量が貯留容量を超えているかチェックする
If Chosui(intLoop, n, i - 1) < CSng(gttypShisetsuPram(n).strChoryuCapacity) Then
    '貯留量が貯留容量を超えていない
'流量の確認
    If QAA1(intLoop, n, i) <= CSng(gttypShisetsuPram(n).strChoryuStart) Then
        '貯留開始容量を超えなかった場合
        Chosui(intLoop, n, i) = Chosui(intLoop, n, i - 1)
        QAA(intLoop, n, i) = QAA1(intLoop, n, i)
    ElseIf IsNumeric(gttypShisetsuPram(n).strChoryuLimit) = False Then
        '流量が貯留開始容量を超えていて、貯留上限流量が設定されていない場合
        GoTo ShisetsuChoryu
    ElseIf QAA1(intLoop, n, i) > CSng(gttypShisetsuPram(n).strChoryuLimit) Then

```

600

650

653

```

intStepSecond
'今回の流量が貯留上限流量を超えた場合
'貯留量は今まで貯留した量+(貯留上限流量(2))-貯留開始量(1)*時間
Chosui(intLoop, n, i) = Chosui(intLoop, n, i - 1) + (CSng(gtypShisetsuPram(n).strChoryuLimit) - CSng(gtypShisetsuPram(n).strChoryuStart)) *
intStepSecond

'放流
If Horyu(n) = 1 Then
  QAA(intLoop, n, i) = CSng(gtypShisetsuPram(n).strChoryuStart)
  Eflu(intLoop, n, i) = QAA1(intLoop, n, i) - CSng(gtypShisetsuPram(n).strChoryuLimit)
Elseif Horyu(n) = 0 Then
  QAA(intLoop, n, i) = CSng(gtypShisetsuPram(n).strChoryuStart) + QAA1(intLoop, n, i) - CSng(gtypShisetsuPram(n).strChoryuLimit)
  Eflu(intLoop, n, i) = 0
End If

Else
'流量が貯留開始容量を超えていて、貯留上限流量以下の場合
'貯留量は今まで貯留した量+(流量-貯留開始量(1))*時間
Chosui(intLoop, n, i) = Chosui(intLoop, n, i - 1) + (QAA1(intLoop, n, i) - CSng(gtypShisetsuPram(n).strChoryuStart)) * intStepSecond
QAA(intLoop, n, i) = CSng(gtypShisetsuPram(n).strChoryuStart)
End If

QAU(n, i) = QAA(intLoop, n, i) - dwf(n)

'貯留量が貯留容量を超えたか確認する
If Chosui(intLoop, n, i) < CSng(gtypShisetsuPram(n).strChoryuCapacity) Then
  GoTo 660
Else
'貯留されなかった分を計算(流量-貯留した量)
QAA2(n, i) = QAA1(intLoop, n, i) - ((CSng(gtypShisetsuPram(n).strChoryuCapacity) - Chosui(intLoop, n, i - 1)) / intStepSecond)
Chosui(intLoop, n, i) = CSng(gtypShisetsuPram(n).strChoryuCapacity)
GoTo 654
End If

Else
'貯留量が貯留容量と同じ
Chosui(intLoop, n, i) = CSng(gtypShisetsuPram(n).strChoryuCapacity)
GoTo 655
End If

```

654

貯留容量を超えたので貯留されなかった分に対して放流の計算を行う

```
If Horyu(n) = 1 Then
  If QAA2(n, i) > CSng(gtypShisetsuPram(n).str3Q) Then
    Eflu(intLoop, n, i) = QAA2(n, i) - CSng(gtypShisetsuPram(n).str3Q)
    QAA(intLoop, n, i) = CSng(gtypShisetsuPram(n).str3Q)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  Else
    Eflu(intLoop, n, i) = 0
    QAA(intLoop, n, i) = QAA2(n, i)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  End If
ElseIf Horyu(n) = 2 Then
  If QAA2(n, i) > CSng(gtypShisetsuPram(n).str3Q) Then
    ByQ(intLoop, n, i) = QAA2(n, i) - CSng(gtypShisetsuPram(n).str3Q)
    QAA(intLoop, n, i) = CSng(gtypShisetsuPram(n).str3Q)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  Else
    ByQ(intLoop, n, i) = 0
    QAA(intLoop, n, i) = QAA2(n, i)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  End If
Else
  Eflu(intLoop, n, i) = 0
  ByQ(intLoop, n, i) = 0
  QAA(intLoop, n, i) = QAA2(n, i)
  QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
End If

GoTo 660

If Horyu(n) = 1 Then
  If QAA1(intLoop, n, i) > Bai(n) * dwfmax(n) Then
    Eflu(intLoop, n, i) = QAA1(intLoop, n, i) - Bai(n) * dwfmax(n)
    QAA(intLoop, n, i) = Bai(n) * dwfmax(n)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  Else
    Eflu(intLoop, n, i) = 0
    QAA(intLoop, n, i) = QAA1(intLoop, n, i)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  End If
```

655

```

ElseIf Horyu(n) = 2 Then
  If QAA1(intLoop, n, i) > Bai(n) * dwfmax(n) Then
    ByQ(intLoop, n, i) = QAA1(intLoop, n, i) - Bai(n) * dwfmax(n)
    QAA(intLoop, n, i) = Bai(n) * dwfmax(n)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  Else
    ByQ(intLoop, n, i) = 0
    QAA(intLoop, n, i) = QAA1(intLoop, n, i)
    QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
  End If
Else
  Eflu(intLoop, n, i) = 0
  ByQ(intLoop, n, i) = 0
  QAA(intLoop, n, i) = QAA1(intLoop, n, i)
  QAU(n, i) = QAA(intLoop, n, i) - dwf(n)
End If

****流下****
If i - Time2 <= 0 Then
  Flow2 = 0
Else
  Flow2 = QAU(n, i - Time2)
End If

If Saki(n) = 0 Then
Else
  If area(Saki(n)) = 0 Then
    inf(intLoop, Saki(n), i) = inf(intLoop, Saki(n), i) + (Flow2 / 10 * 360)
  Else
    inf(intLoop, Saki(n), i) = inf(intLoop, Saki(n), i) + (Flow2 / area(Saki(n)) * 360)
  End If
End If

****ノイノズ****
If Horyu(n) = 2 Then
  If i - Time3 <= 0 Then
    Flow3 = 0
  Else
    Flow3 = ByQ(intLoop, n, i - Time3)
  End If

  If area(Bysaki(n)) = 0 Then

```

660

665


```

        inf(intLoop, Bysaki(n), i) = inf(intLoop, Bysaki(n), i) + (Flow3 / 10 * 360)
    Else
        inf(intLoop, Bysaki(n), i) = inf(intLoop, Bysaki(n), i) + (Flow3 / area(Bysaki(n)) * 360)
    End If
End If
'670
Next i
DoEvents
Next n

Next intLoop
'処理ステップ管理ファイルを更新する
Open App.Path & "\Case¥" & "¥" & g_CName & "\¥CalcStep.Par" For Output As #1
Print #1, 2 '流量計算実行済み
Close #1

Sub_FrowCalc_Exit:
Exit Sub
Sub_FrowCalc_Err:
MsgBox Err.Description
Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation] : " & "Sub_FrowCalc", "流量計算処理にてエラーが発生しました。")
Resume Sub_FrowCalc_Exit

End Sub

*****
' 汚濁負荷計算処理
*****
' パブリックエリアの変数データを元に汚濁負荷計算を行う
*****
Public Sub Sub_PollutionCalc(in_CalcChk0 As Integer, in_CalcChk1 As Integer)
On Error GoTo Sub_PollutionCalc_Err
On Error GoTo 0

Dim i As Integer
Dim j As Integer
Dim n As Integer
Dim TTC3 As Integer
Dim bss As Single
        TTC2 * 60

```

```

Dim af As Single
Dim bs As Single
Dim PS0 As Single
Dim LWPP As Single

Dim MM(5) As Integer
Dim NN(6) As Integer

Dim CFUKA As Single
Dim Time2 As Integer
Dim Time3 As Integer
Dim w_CC As Single

Dim intLoop As Integer 'ループ用カウンタ
Dim intCount As Integer '降雨毎の最大計算ステップをカウントするためのワーク
Dim intStepSecond As Integer '1ステップあたりの秒数

intStepSecond = CInt(gstrStepMinute) * 60 '1ステップあたりの秒数を設定

'ループに対応するため、格納用の変数の範囲を最初に宣言
maxkkou = UBound(gtypKouuData(1).strKouuData) '最大計算ステップ数は降雨データを格納している変数の配列数
TC2 = gstrStepMinute '降雨時間(分/1ステップ)

ReDim LWS3(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single
ReDim LWP3(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single
ReDim Eload(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single
ReDim EloadB(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single
ReDim TFUKA(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single

'管からの流出負荷量(g/s)
'管さよ+路面負荷(g/s) (遮集)
'貯留負荷(トータル)

'他の変数と同様の参照方法で動かせるように宣言しているが、
' 対策施設の場合のみデータが入るため、無駄にメモリを使用しています。
' 件数次第ではかなりシステムへの影響が大きくなります。

ReDim ChoT(UBound(gtypKouuData), 5, RYUSU, maxkkou) As Single

'降雨毎にループする
MenuF.lblCalcStatus.Visible = True
For intLoop = 1 To UBound(gtypKouuData)

    MenuF.lblCalcStatus.Caption = "汚濁負荷計算実行中です" & vbCrLf & "(" & intLoop & "/" & UBound(gtypKouuData) & "件目)"
    DoEvents

'ローカル変数をループ毎に初期化

```

```

TTC3 = 0
bss = 0
af = 0
bs = 0
LWPP = 0

CFUKA = 0
Time2 = 0
Time3 = 0
w_CC = 0

読み込んだ行ごとのステップ数
maxkkou = gtypKouuData(intLoop).lngStepCount

'プロシージャ内でのみ使用する変数なので、ループするたびに初期化する
ReDim PS(maxkkou) As Single
ReDim LWS(5, RYUSU, maxkkou) As Single
ReDim LWS2(5, RYUSU, maxkkou) As Single

ReDim LWP(5, RYUSU, maxkkou) As Single
'管渠からの流出負荷量(g/s)

ReDim PP(5, RYUSU, maxkkou) As Single
'管渠堆積負荷量
ReDim PPP(5, RYUSU, maxkkou) As Single
ReDim BupP(5, RYUSU, maxkkou) As Single
ReDim EupP(5, RYUSU, maxkkou) As Single
ReDim EloadS(5, RYUSU, maxkkou) As Single
ReDim EloadP(5, RYUSU, maxkkou) As Single
ReDim ChoP(5, RYUSU, maxkkou) As Single
ReDim ChoS(5, RYUSU, maxkkou) As Single
'貯留負荷(管内)
'貯留負荷(路面)

ReDim GG(RYUSU) As Single
ReDim DL(5, RYUSU) As Single
ReDim PP0(5, RYUSU) As Single

*****路面流出負荷量計算*****
' RC(n) '限界降雨強度
' TC2 '一つの流量の時間間隔(分) 通常 5分
TTC3 = TC2 * 60

*****路面流出負荷量計算*****
For j = 1 To 5
'5 は水質項目数(BOD,COD,SS,TP,TN)

```

```

Select Case j
Case 1, 2, 3 'BOD,COD,SSを計算
If in_CalcChk0 = ssCBUnchecked Then
GoTo JLoop_End
End If
Case 4, 5 'TPT-Nを計算
If in_CalcChk1 = ssCBUnchecked Then
GoTo JLoop_End
End If
End Select

For n = 1 To RYUSU
'不浸透面積率(Imp)にて判定
If Imp(n) = 0 Then
LWS(j, n, 1) = 0
PS(1) = 0
Else
LWS(j, n, 1) = 0
af = 1000 * CC * Imp(n) * area(n)
bs = -Pub_a(j) - af * Pub_PSO(j) / TTC3
'af=αのこと。 CC : 不浸透域での負荷発生割合(通常1)
'流域面積(ha)にて判定
If area(n) = 0 Then
LWS(j, n, 1) = 0
PS(1) = 0
Else
'加算前インフローを使用する
If inf0(intLoop, n, 1) - Pub_RC(j) < 0 Then
PS(1) = -2 * TTC3 * bs / (2 * af)
LWS(j, n, 1) = 0
Else
PS(1) = -2 * TTC3 * bs / (2 * af + Pub_CR(j) * TTC3 * (inf0(intLoop, n, 1) - Pub_RC(j)) * area(n) / 3.6)
If PS(1) < 0 Then
PS(1) = 0
End If
LWS(j, n, 1) = 2 * Pub_a(j) - 2 * af * (PS(1) - Pub_PSO(j)) / TTC3
If LWS(j, n, 1) < 0 Then
LWS(j, n, 1) = 0
End If
End If
End If
End If

```

```

For i = 2 To maxkkou
  If Impp(n) = 0 Then
    LWS(j, n, i) = 0
    PS(i) = 0
  Else
    If area(n) = 0 Then
      LWS(j, n, i) = 0
      PS(i) = 0
    Else
      '加算前インフローを使用する
      bss = -Pub_a(i) - af * PS(i - 1) / TTC3 + LWS(j, n, i - 1) / 2
      If inf0(intLoop, n, i) - Pub_RC(j) < 0 Then
        PS(i) = -2 * TTC3 * bss / (2 * af)
        LWS(j, n, i) = 0
      Else
        PS(i) = -2 * TTC3 * bss / (2 * af + Pub_CR(j) * TTC3 * (inf0(intLoop, n, i) - Pub_RC(j)) * area(n) / 3.6)
        If PS(i) < 0 Then
          PS(i) = 0
        End If
        LWS(j, n, i) = Round(2 * Pub_a(i) - 2 * af * (PS(i) - PS(i - 1)) / TTC3 - LWS(j, n, i - 1), 2)
        If LWS(j, n, i) < 0 Then
          LWS(j, n, i) = 0
        End If
      End If
    End If
  End If
End If
Next i
Next n

*****管渠流出負荷量計算*****
'管渠負荷計算式定数
MM(1) = 2
MM(2) = 2
MM(3) = 1
MM(4) = 2
MM(5) = 2
NN(1) = 0
NN(2) = 0
NN(3) = 1
NN(4) = 0

```

```

NN(6) = 0
For n = 1 To RYUSU
  PP0(j, n) = Pub_RN(j) * dwf0(n) * 86400 * 0.5 '初期管内堆積負荷量 (g) 晴天時汚水量(加算前) * 24 時間 * 0.5
  GG(n) = area(n) * Pub_GG(j) '限界流量(m3/s)
  DL(j, n) = Pub_RN(j) * dwf0(n) '晴天時発生負荷量 (g/s)
Next n

For n = 1 To RYUSU
  If Saki(n) = 0 Then '流下先ブロックにて判定
  Else
    PP0(j, Saki(n)) = PP0(j, Saki(n)) + PP0(j, n)
    GG(Saki(n)) = GG(Saki(n)) + GG(n)
    DL(j, Saki(n)) = DL(j, Saki(n)) + DL(j, n)
  End If

  If PP0(j, n) = 0 Then
    LWP(j, n, 1) = BupP(j, n, 1)
    PP(j, n, 1) = 0
  Else
    'SS 以外は 1.2 倍
    If j = 3 Then
      CFUKA = Pub_RN(j) / (PP0(j, n) * dwf(n))
    Else
      CFUKA = (Pub_RN(j) / (PP0(j, n) ^ 2)) * 1.2
    End If

    PP(j, n, 1) = PP0(j, n)
    LWP(j, n, 1) = CFUKA * PP(j, n, 1) ^ MM(j) * QAA(intLoop, n, 1) ^ NN(j) * QAA(intLoop, n, 1) - GG(n)
    PP(j, n, 2) = -1 * TTC3 / 2 * LWP(j, n, 1) + PP0(j, n) - 1 * TTC3 / 2 * (-1 * DL(j, n)) - EupP(j, n, 1) * TTC3 / 2 + BupP(j, n, 1) * TTC3 / 2
  End If

  If IsNumeric(gtYpShisetsuPram(n).strNo) = True Then
    If Chosui(intLoop, n, 1) = 0 Then
      ChoP(j, n, 1) = 0
      ChoS(j, n, 1) = 0
    Else
      ChoP(j, n, 1) = LWP(j, n, 1) / (QAA1(intLoop, n, 1) + Chosui(intLoop, n, 1) / intStepSecond) * (Chosui(intLoop, n, 1) / intStepSecond)
      ChoS(j, n, 1) = LWS(j, n, 1) / (QAA1(intLoop, n, 1) + Chosui(intLoop, n, 1) / intStepSecond) * (Chosui(intLoop, n, 1) / intStepSecond)
      LWP(j, n, 1) = LWP(j, n, 1) / (QAA1(intLoop, n, 1) + Chosui(intLoop, n, 1) / intStepSecond) * QAA1(intLoop, n, 1)
    End If
  End If

```

```

LWS(j, n, 1) = LWS(j, n, 1) / (QAA1(intLoop, n, 1) + Chosui(intLoop, n, 1) / intStepSecond) * QAA1(intLoop, n, 1)
End If
End If
ChoT(intLoop, j, n, 1) = ChoP(j, n, 1) + ChoS(j, n, 1)
End If
End If

*****越流負荷量および遮集負荷の分け*****
If QAA1(intLoop, n, 1) = 0 Then
  LWS3(intLoop, j, n, 1) = LWS(j, n, 1)
  LWP3(intLoop, j, n, 1) = LWP(j, n, 1)
Else
  If Horyu(n) = 1 Then
    EloadS(j, n, 1) = LWS(j, n, 1) / QAA1(intLoop, n, 1) * Eflu(intLoop, n, 1)
    EloadP(j, n, 1) = LWP(j, n, 1) / QAA1(intLoop, n, 1) * Eflu(intLoop, n, 1)
  LWS3(intLoop, j, n, 1) = LWS(j, n, 1) - EloadS(j, n, 1)
  LWP3(intLoop, j, n, 1) = LWP(j, n, 1) - EloadP(j, n, 1)
  ElseIf Horyu(n) = 2 Then
    EloadS(j, n, 1) = LWS(j, n, 1) / QAA1(intLoop, n, 1) * ByQ(intLoop, n, 1)
    EloadP(j, n, 1) = LWP(j, n, 1) / QAA1(intLoop, n, 1) * ByQ(intLoop, n, 1)
  LWS3(intLoop, j, n, 1) = LWS(j, n, 1) - EloadS(j, n, 1)
  LWP3(intLoop, j, n, 1) = LWP(j, n, 1) - EloadP(j, n, 1)
  Else
    LWS3(intLoop, j, n, 1) = LWS(j, n, 1)
    LWP3(intLoop, j, n, 1) = LWP(j, n, 1)
  End If
End If

*****地表面負荷は時間遅れで下流に受け渡し、管渠負荷は堆積負荷から差引く越流負荷を算出*****
LWS2(j, n, 1) = 0

Time2 = Round((RTJ(n) + 0.00001) / TC2, 0)
Time3 = Round(ByTime(n) / TC2, 0)

If 1 - Time2 > 0 Then
  If Saki(n) = 0 Then
    Else
      EupP(j, Saki(n), 1) = EupP(j, Saki(n), 1) + EloadP(j, n, 1 - Time2) + ChoP(j, n, 1 - Time2)
      LWS(j, Saki(n), 1) = LWS(j, Saki(n), 1) + LWS3(intLoop, j, n, 1 - Time2)
    End If
  End If
End If

```

```

If Horyu(n) = 2 Then
  If 1 - Time3 > 0 Then
    LWS(j, Bysaki(n), 1) = LWS(j, Bysaki(n), 1) + EloadS(j, n, 1 - Time3)
    BupP(j, Bysaki(n), 1) = BupP(j, Bysaki(n), 1) + EloadP(j, n, 1 - Time3)
  End If
End If

Eload(intLoop, j, n, 1) = EloadS(j, n, 1) + EloadP(j, n, 1)
TFUKA(intLoop, j, n, 1) = LWS3(intLoop, j, n, 1) + LWP3(intLoop, j, n, 1)

For i = 2 To maxkkou
  If PPO(j, n) = 0 Then
    LWP(j, n, i) = BupP(j, n, i)
    PP(j, n, i) = 0
  Else
    Select Case MM(j)
      Case 1
        ***** 1次式を解く場合*****
        LWP(j, n, i) = CFUKA * PPP(j, n, i) ^ MM(j) * QAA(intLoop, n, i) ^ NN(j) * (QAA(intLoop, n, i) - GG(n))
        If LWP(j, n, i) < 0 Then
          LWP(j, n, i) = 0
        End If
        LWPP = Round(LWP(j, n, i), 1)
        w_CC = -1 * TTC3 / 2 * (2 * DL(j, n) - LWPP) - PPP(j, n, i)
        PP(j, n, i) = -w_CC / (1 + 0.5 * TTC3 * CFUKA * QAA(intLoop, n, i) * (QAA(intLoop, n, i) - GG(n)))
        If i < maxkkou Then
          PPP(j, n, i + 1) = Round(PP(j, n, i), 0)
        End If
      Case 2
        ***** 2次方程式を解く場合*****
        LWP(j, n, i) = CFUKA * PP(j, n, i) ^ MM(j) * QAA1(intLoop, n, i) ^ NN(j) * (QAA1(intLoop, n, i) - GG(n))
        If LWP(j, n, i) < 0 Then
          LWP(j, n, i) = 0
        ElseIf LWP(j, n, i) >= PP(j, n, i) / TTC3 Then
          LWP(j, n, i) = PP(j, n, i) / TTC3
        End If
      Case 3
        If i < maxkkou Then
          PP(j, n, i + 1) = PP(j, n, i) + (DL(j, n) - LWP(j, n, i) - EupP(j, n, i) + BupP(j, n, i)) * TTC3
        End If
    End Select
  End If
Next i

```



```

If PPP(j, n, i + 1) < 0 Then
  PP(j, n, i + 1) = 0
End If
End If
End Select
End If

If IsNumeric(gtypShisetsuPram(n).strNo) = True Then
  If gtypShisetsuPram(n).strChoryuCapacity > 0 Then
    If Chosui(intLoop, n, i) - Chosui(intLoop, n, i - 1) = 0 Then
      ChoP(j, n, i) = 0
      ChoS(j, n, i) = 0
    Else
      ChoP(j, n, i) = LWP(j, n, i) / QAA1(intLoop, n, i) * ((Chosui(intLoop, n, i) - Chosui(intLoop, n, i - 1)) / intStepSecond)
      ChoS(j, n, i) = LWS(j, n, i) / QAA1(intLoop, n, i) * ((Chosui(intLoop, n, i) - Chosui(intLoop, n, i - 1)) / intStepSecond)
      LWP(j, n, i) = LWP(j, n, i) - ChoP(j, n, i)
      LWS(j, n, i) = LWS(j, n, i) - ChoS(j, n, i)
      ChoT(intLoop, j, n, i) = ChoP(j, n, i) + ChoS(j, n, i)
    End If
  End If
End If

*****越流負荷量および遮集負荷、バイパス負荷の分け*****
If QAA1(intLoop, n, 1) = 0 Then
  LWS3(intLoop, j, n, 1) = LWS(j, n, 1)
  LWP3(intLoop, j, n, 1) = LWP(j, n, 1)
Else
  If Horyu(n) = 1 Then
    EloadS(j, n, i) = LWS(j, n, i) / QAA1(intLoop, n, i) * Eflu(intLoop, n, i)
    EloadP(j, n, i) = LWP(j, n, i) / QAA1(intLoop, n, i) * Eflu(intLoop, n, i)

    LWS3(intLoop, j, n, i) = LWS(j, n, i) - EloadS(j, n, i)
    LWP3(intLoop, j, n, i) = LWP(j, n, i) - EloadP(j, n, i)
  ElseIf Horyu(n) = 2 Then
    EloadS(j, n, i) = LWS(j, n, i) / QAA1(intLoop, n, i) * ByQ(intLoop, n, i)
    EloadP(j, n, i) = LWP(j, n, i) / QAA1(intLoop, n, i) * ByQ(intLoop, n, i)

    LWS3(intLoop, j, n, i) = LWS(j, n, i) - EloadS(j, n, i)
    LWP3(intLoop, j, n, i) = LWP(j, n, i) - EloadP(j, n, i)
  Else

```

'1500

'1550

```

LWS3(intLoop, j, n, i) = LWS(j, n, i)
LWP3(intLoop, j, n, i) = LWP(j, n, i)
End If
End If

*****地表面負荷は時間遅れで下流に受け渡し、管渠負荷は堆積負荷から差引く越流負荷を算出*****
If i - Time2 <= 0 Then
  LWS2(j, n, i) = 0
Else
  LWS2(j, n, i) = LWS3(intLoop, j, n, i - Time2)
  If Saki(m) = 0 Then
    Else
      EupP(j, Saki(n), i) = EupP(j, Saki(m), i) + EloadP(j, n, i - Time2) + ChoP(j, n, i - Time2)
      LWS(j, Saki(m), i) = LWS2(j, n, i) + LWS(j, Saki(n), i)
    End If
  End If
End If

If Horyu(n) = 2 Then
  If i - Time3 >= 0 Then
    LWS(j, Bysaki(n), i) = LWS(j, Bysaki(m), i) + EloadS(j, n, i - Time3)
    BupP(j, Bysaki(n), i) = BupP(j, Bysaki(m), i) + EloadP(j, n, i - Time3)
  End If
End If

If Horyu(n) = 1 Then
  Eload(intLoop, j, n, i) = EloadS(j, n, i) + EloadP(j, n, i)
ElseIf Horyu(n) = 2 Then
  EloadB(intLoop, j, n, i) = EloadS(j, n, i) + EloadP(j, n, i)
Else
  Eload(intLoop, j, n, i) = 0
End If

TFUKA(intLoop, j, n, i) = LWS3(intLoop, j, n, i) + LWP3(intLoop, j, n, i)
Next i
Next n
JLoop_End:
Next j

Next intLoop
'処理場計算
Call subSyorijyouCalc

```

```

'雨水吐き・ポンプ場等計算
Call subUsuiCalc

Sub_Sub_PollutionCalc_Exit:
Exit Sub
Sub_PollutionCalc_Err:
MsgBox Err.Description
Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation]:" & "Sub_PollutionCalc", "汚濁負荷計算処理にてエラーが発生しました。")
Resume Sub_Sub_PollutionCalc_Exit
End Sub

*****
'計算が完了したら結果の合計と平均を計算する
'引数:xintPollutionFlg
, (0:流量計算のみ 1:汚濁負荷計算も行った)
*****
Public Sub subCulcSummary_Frow(xintPollutionFlg As Integer)
Dim i As Integer
Dim j As Integer
Dim k As Integer
Dim n As Integer
Dim intKouuCount As Integer '降雨数
Dim intDataCount As Integer '最大スラップ数

MenuF.lblCalcStatus.Caption = "計算結果を集計しています。"
DoEvents

intKouuCount = UBound(gttypKouuData)
intDataCount = UBound(gttypKouuData(1).strKouuData)

'結果格納用の変数の初期化
ReDim gSummary_YR(1, RYUSU, intDataCount)
ReDim gSummary_inf0(1, RYUSU, intDataCount)
ReDim gSummary_inf(1, RYUSU, intDataCount)
ReDim gSummary_Eflu(1, RYUSU, intDataCount)
ReDim gSummary_ByQ(1, RYUSU, intDataCount)
ReDim gSummary_QAA(1, RYUSU, intDataCount)
ReDim gSummary_QAA1(1, RYUSU, intDataCount)
ReDim gSummary_LWS3(1, 5, RYUSU, intDataCount)
ReDim gSummary_LWP3(1, 5, RYUSU, intDataCount)
ReDim gSummary_Eload(1, 5, RYUSU, intDataCount)

'放流量
'トイレパルス量
'雨水+汚水
'雨水+汚水
'管渠からの流出負荷量(g/s)

```

```

ReDim gSummary_EloadB(1, 5, RYUSU, intDataCount)
ReDim gSummary_TFUKA(1, 5, RYUSU, intDataCount) '管きよ + 路面負荷(g/s) (遮集)

'合計を計算する
'降雨数でループ
For i = 1 To intKouuCount
    MenuFileCalcStatus.Caption = "計算結果を集計しています。" & vbCrLf & "(" & i & "/" & intKouuCount & "件目)"
    DoEvents

    '流域数でループ
    For n = 1 To RYUSU
        'ステップでループ
        For k = 1 To intDataCount
            '流域とステップで各降雨のデータを格納
            gSummary_YR(0, n, k) = gSummary_YR(0, n, k) + YR(i, n, k)
            gSummary_inf(0, n, k) = gSummary_inf(0, n, k) + inf(i, n, k)
            gSummary_inf0(0, n, k) = gSummary_inf0(0, n, k) + inf0(i, n, k)
            gSummary_Eflu(0, n, k) = gSummary_Eflu(0, n, k) + Eflu(i, n, k)
            gSummary_ByQ(0, n, k) = gSummary_ByQ(0, n, k) + ByQ(i, n, k)
            gSummary_QAA(0, n, k) = gSummary_QAA(0, n, k) + QAA(i, n, k)
            gSummary_QAA1(0, n, k) = gSummary_QAA1(0, n, k) + QAA1(i, n, k)

            '汚濁負荷計算を行っていた場合
            If xintPollutionFlg = 1 Then
                For j = 1 To 5
                    gSummary_LWS3(0, j, n, k) = gSummary_LWS3(0, j, n, k) + LWS3(i, j, n, k)
                    gSummary_LWP3(0, j, n, k) = gSummary_LWP3(0, j, n, k) + LWP3(i, j, n, k)
                    gSummary_Eload(0, j, n, k) = gSummary_Eload(0, j, n, k) + Eload(i, j, n, k)
                    gSummary_EloadB(0, j, n, k) = gSummary_EloadB(0, j, n, k) + EloadB(i, j, n, k)
                    gSummary_TFUKA(0, j, n, k) = gSummary_TFUKA(0, j, n, k) + TFUKA(i, j, n, k)
                Next j
            End If
        Next k
    Next n
Next i

End Sub

'*****
' 処理場計算
' 処理場計算を行う
' *****

```

```

*****
Public Sub subSyorijyouCalc()
On Error GoTo subSyorijyouCalc_Err
On Error GoTo 0
Dim sngTotalRyo As Single '高級処理量と簡易処理量の合計
Dim sngKoukyuuRyo As Single '高級処理量
Dim sngKaniRyo As Single '簡易処理量
Dim sngChoryuMax As Single '貯留容量
Dim sngkokuuyuuritsu(5) As Single '高級処理率
Dim sngkaniritsu(5) As Single '簡易処理率
Dim w_Quality As Integer '水質基準 index

Dim sngOver As Single '高級処理量と簡易処理量を超えた水量が流入したときに、
'簡易処理を行う水量を求めるためのワーク変数

Dim chkFlg As Integer '判定
Dim i As Integer
Dim intRyuukiNo As Integer '処理場の流域番号
Dim RAINNUM As Integer '降雨番号
Dim SYORINUM As Integer '処理場の数

Dim intKouuCount As Integer '降雨数
Dim intRyuukiCount As Integer '流域数
Dim intStepSecond As Integer '1ステップあたりの秒数

Dim sngChoryuWork As Single '貯留量の判定用に貯留量を一時保存する変数

intKouuCount = UBound(gttypKouuData)
intStepSecond = CInt(gstrStepMinute) * 60 '1 ステップあたりの秒数を設定
maxkkou = UBound(gttypKouuData(1).strKouuData) '最大計算ステップ数は降雨データを格納している変数の配列数

'計算結果格納用のグローバル変数を初期化
ReDim gsngKokyu(UBound(gttypSyorijyouPram), intKouuCount, maxkkou)
ReDim gsngChoryuRyuryo(UBound(gttypSyorijyouPram), intKouuCount, maxkkou)
ReDim gsngChoryu(UBound(gttypSyorijyouPram), intKouuCount, maxkkou)
ReDim gsngKani(UBound(gttypSyorijyouPram), intKouuCount, maxkkou)
ReDim gsngChokuhou(UBound(gttypSyorijyouPram), intKouuCount, maxkkou)

ReDim gsngSuishitu(UBound(gttypSyorijyouPram), intKouuCount, maxkkou, 5)
ReDim gintHouryuCount(UBound(gttypSyorijyouPram))

ReDim gsngKokyu_Fuka(UBound(gttypSyorijyouPram), intKouuCount, maxkkou, 5)

```

ReDim gsngChoryu_Fuka(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '貯留負荷量
 ReDim gsngKani_Fuka(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '簡易負荷量
 ReDim gsngChokuhou_Fuka(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '直放負荷量

 ReDim gsngKokyu_Houryu(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '高級放流量
 ReDim gsngChoryu_Houryu(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '貯留放流量
 ReDim gsngKani_Houryu(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '簡易放流量
 ReDim gsngChokuhou_Houryu(UBound(gtppSyorijyouPram), intKouuCount, maxkkou, 5) '直放放流量

 '年間
 ReDim gsngKokyu_Year(UBound(gtppSyorijyouPram)) '高級量
 ReDim gsngChoryu_Year(UBound(gtppSyorijyouPram)) '貯留量
 ReDim gsngKani_Year(UBound(gtppSyorijyouPram)) '簡易量
 ReDim gsngChokuhou_Year(UBound(gtppSyorijyouPram)) '直放量

 ReDim gsngKokyu_Fuka_Year(UBound(gtppSyorijyouPram), 5) '高級負荷量
 ReDim gsngChoryu_Fuka_Year(UBound(gtppSyorijyouPram), 5) '貯留負荷量
 ReDim gsngKani_Fuka_Year(UBound(gtppSyorijyouPram), 5) '簡易負荷量
 ReDim gsngChokuhou_Fuka_Year(UBound(gtppSyorijyouPram), 5) '直放負荷量

 ReDim gsngKokyu_Houryu_Year(UBound(gtppSyorijyouPram), 5) '高級放流量
 ReDim gsngChoryu_Houryu_Year(UBound(gtppSyorijyouPram), 5) '貯留放流量
 ReDim gsngKani_Houryu_Year(UBound(gtppSyorijyouPram), 5) '簡易放流量
 ReDim gsngChokuhou_Houryu_Year(UBound(gtppSyorijyouPram), 5) '直放放流量

 ReDim 放流回数(RYUSU)
 ReDim 年間合計高級量(RYUSU) As Single
 ReDim 年間合計貯留量(RYUSU) As Single
 ReDim 年間合計簡易量(RYUSU) As Single
 ReDim 年間合計直放量(RYUSU) As Single

 ReDim 年間合計高級負荷量(RYUSU, 5) As Single
 ReDim 年間合計貯留負荷量(RYUSU, 5) As Single
 ReDim 年間合計簡易負荷量(RYUSU, 5) As Single
 ReDim 年間合計直放負荷量(RYUSU, 5) As Single

 ReDim 年間合計高級放流負荷量(RYUSU, 5) As Single
 ReDim 年間合計貯留放流負荷量(RYUSU, 5) As Single
 ReDim 年間合計簡易放流負荷量(RYUSU, 5) As Single
 ReDim 年間合計直放放流負荷量(RYUSU, 5) As Single

 ReDim 高級量(UBound(gtppSyorijyouPram), intKouuCount, maxkkou) As Single

```

ReDim 貯留流量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou) As Single
ReDim 貯留量(UBound(gtypSyorijyouPram), intKouuCount) As Single 'これだけ 1 回の雨の累計の貯留量なので、ステップは関係ありません
ReDim 簡易量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou) As Single
ReDim 直放量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou) As Single
ReDim 水質(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single

ReDim 高級負荷量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 貯留負荷量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 簡易負荷量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 直放負荷量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single

ReDim 合計高級量(UBound(gtypSyorijyouPram), intKouuCount) As Single
ReDim 合計貯留量(UBound(gtypSyorijyouPram), intKouuCount) As Single
ReDim 合計簡易量(UBound(gtypSyorijyouPram), intKouuCount) As Single
ReDim 合計直放量(UBound(gtypSyorijyouPram), intKouuCount) As Single

ReDim 高級放流量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 貯留放流量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 簡易放流量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single
ReDim 直放放流量(UBound(gtypSyorijyouPram), intKouuCount, maxkkou, 5) As Single

ReDim 合計高級放流量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計貯留放流量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計簡易放流量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計直放放流量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single

ReDim 合計高級負荷量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計貯留負荷量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計簡易負荷量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single
ReDim 合計直放負荷量(UBound(gtypSyorijyouPram), intKouuCount, 5) As Single

'すべての降雨で計算
For RAINNUM = 1 To intKouuCount
    MenuFblCalcStatus.Caption = "処理場計算実行中です" & vbCrLf & "(" & RAINNUM & "/" & intKouuCount & "件目)"
    DoEvents

    17000
    *****処理場計算*****
    sngTotalRyo = sngKoukyuuRyo + sngKaniRyo
    '処理場の処理だけ実行する
    For SYORINUM = 1 To UBound(gtypSyorijyouPram)

```

```

intRyuukiNo = gtypSyorijyouPram(SYORINUM).strNo
sngKoukyuuRyo = gtypSyorijyouPram(SYORINUM).strKoukyuu
sngKaniRyo = gtypSyorijyouPram(SYORINUM).strKani
sngTotalRyo = sngKoukyuuRyo + sngKaniRyo
'2012/04/01 水質基準種類追加
For w_Quality = 1 To 5
    sngkoukyuuritsu(w_Quality) = gtypSyorijyouPram(SYORINUM).strKoukyuuritsu(w_Quality)
    sngkaniritsu(w_Quality) = gtypSyorijyouPram(SYORINUM).strKaniritsu(w_Quality)
Next w_Quality
sngChoryuWork = 0

'貯留量の計算を追加
sngChoryuMax = gtypSyorijyouPram(SYORINUM).strChoryuCapacity

GoTo 17030

17005 Next SYORINUM

Next RAINNUM

'すべての降雨の計算が終わったら終了
GoTo subSyorijyouCalc_Exit

17030 *****処理場計算*****
17033 *****流量と負荷量の振り分け(高級・簡易・直接放流)*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    '2012/04/01 水質基準種類追加
    For w_Quality = 1 To 5
        水質(SYORINUM, RAINNUM, i, w_Quality) = TFUKA(RAINNUM, w_Quality, intRyuukiNo, i) / QAA(RAINNUM, intRyuukiNo, i)
    Next w_Quality

    If QAA(RAINNUM, intRyuukiNo, i) <= sngKoukyuuRyo Then
        '流入量が高級処理量より少ない場合
        高級量(SYORINUM, RAINNUM, i) = Format(QAA(RAINNUM, intRyuukiNo, i) * intStepSecond, "#.##0.0")
        簡易量(SYORINUM, RAINNUM, i) = 0
        直放量(SYORINUM, RAINNUM, i) = 0

    '2012/04/01 水質基準種類追加
    For w_Quality = 1 To 5

```



```

*****処理場流入負荷量*****
高級負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(高級量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "#,##0.0")
簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) = 0
直放負荷量(SYORINUM, RAINNUM, i, w_Quality) = 0

簡易放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkaniritsu(w_Quality) / 100),
"#,##0.0")

直放放流量(SYORINUM, RAINNUM, i, w_Quality) = 直放負荷量(SYORINUM, RAINNUM, i, w_Quality)
Next w_Quality

ElseIf (sngKoukyuuRyo < QAA(RAINNUM, intRyuukiNo, i)) And (QAA(RAINNUM, intRyuukiNo, i) <= sngTotalRyo) Then
'流入量が高級処理量より多く、高級処理量+簡易処理量より少ない場合
高級量(SYORINUM, RAINNUM, i) = Format(sngKoukyuuRyo * intStepSecond, "#,##0.0")

'貯留容量に到達するまでは貯留する
If 貯留量(SYORINUM, RAINNUM) < sngChoryuMax Then
'貯留量と貯留量の計算を修正
貯留量(SYORINUM, RAINNUM, i) = (QAA(RAINNUM, intRyuukiNo, i) - sngKoukyuuRyo) * intStepSecond
'全部貯留した場合の貯留量を計算
sngChoryuWork = Format(貯留量(SYORINUM, RAINNUM) + 貯留量(SYORINUM, RAINNUM, i), "#,##0.0")

If sngChoryuWork >= sngChoryuMax Then
'全部貯留した場合に、貯留量の上限を超える場合
貯留量は貯留上限-現在の貯留量
貯留量(SYORINUM, RAINNUM, i) = sngChoryuMax - 貯留量(SYORINUM, RAINNUM)
'超えた分は簡易処理に回る
簡易量(SYORINUM, RAINNUM, i) = Format(sngChoryuWork - sngChoryuMax, "#,##0.0")
貯留量を貯留容量と同じに設定する
貯留量(SYORINUM, RAINNUM) = sngChoryuMax
Else
'貯留容量に到達しない場合
'現在の貯留量に貯留量を加算する
貯留量(SYORINUM, RAINNUM) = Format(貯留量(SYORINUM, RAINNUM) + 貯留量(SYORINUM, RAINNUM, i), "#,##0.0")
'貯留量が貯留容量を超えていたら簡易処理の対象量は0
簡易量(SYORINUM, RAINNUM, i) = 0

End If
Else
'貯留容量がいっぱいの場合

```

165000

```

簡易量(SYORINUM, RAINNUM, i) = Format((QAA(RAINNUM, intRyuukiNo, i) - sngKoukyuuRyo) * intStepSecond, "###0.0")
End If

直放量(SYORINUM, RAINNUM, i) = 0

2012/04/01 水質基準種類追加
For w_Quality = 1 To 5
  高級負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(高級量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "###0.0")
  貯留負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(貯留流量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "###0.0")
  簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(簡易量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "###0.0")
  直放負荷量(SYORINUM, RAINNUM, i, w_Quality) = 0

  貯留放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(貯留負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkoukyuuritsu(w_Quality) / 100),
"###0.0")

  簡易放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkaniritsu(w_Quality) / 100),
"###0.0")

  直放放流量(SYORINUM, RAINNUM, i, w_Quality) = 直放負荷量(SYORINUM, RAINNUM, i, w_Quality)
Next w_Quality
ElseIf sngTotalRyo < QAA(RAINNUM, intRyuukiNo, i) Then
  流入量が高級処理量と簡易処理量の合計より多い場合
  高級量(SYORINUM, RAINNUM, i) = Format(sngKoukyuuRyo * intStepSecond, "###0.0")

  貯留容量に到達するまでは貯留する
  If 貯留量(SYORINUM, RAINNUM) < sngChoryuMax Then
    '貯留量と貯留流量の計算を修正
    貯留量(SYORINUM, RAINNUM, i) = (QAA(RAINNUM, intRyuukiNo, i) - sngKoukyuuRyo) * intStepSecond
    '全部貯留した場合の貯留量を計算
    sngChoryuWork = Format(貯留量(SYORINUM, RAINNUM) + 貯留流量(SYORINUM, RAINNUM, i), "###0.0")

  If sngChoryuWork >= sngChoryuMax Then
    '貯留流量は貯留上限-現在の貯留量
    貯留量(SYORINUM, RAINNUM, i) = sngChoryuMax - 貯留量(SYORINUM, RAINNUM)
    '貯留量が貯留容量を超えたら超えた分が簡易処理と放流に回る
    sngOver = Format(sngChoryuWork - sngChoryuMax, "###0.0")
    '貯留量を貯留容量と同じに設定する
    貯留量(SYORINUM, RAINNUM) = sngChoryuMax
  Else
    '貯留量が貯留容量を超えていなかった場合は簡易処理の対象水量は 0
    sngOver = 0
    '貯留容量に到達していない場合は、貯留量に流入量-高級処理量を加算する
    貯留量(SYORINUM, RAINNUM) = 貯留量(SYORINUM, RAINNUM) + Format((QAA(RAINNUM, intRyuukiNo, i) - sngKoukyuuRyo) * intStepSecond,
"###0.0")

```

```

End If
Else
'貯留容量がいっぱいの場合
sngOver = Format((QAA(RAINNUM, intRyuunkiNo, i) - sngKouyuuRyo) * intStepSecond, "#,##0.0")
End If

'簡易処理または放流が必要か
If sngOver > 0 Then
'簡易処理が必要な場合
If sngOver <= Format(sngKaniRyo * intStepSecond, "#,##0.0") Then
'超過分が簡易処理量より少なかつた場合
簡易量(SYORINUM, RAINNUM, i) = Format(sngOver, "#,##0.0")
直放量(SYORINUM, RAINNUM, i) = 0
Else
'超過分が簡易処理量よりも多かつた場合
簡易量(SYORINUM, RAINNUM, i) = Format(sngKaniRyo * intStepSecond, "#,##0.0")
直放量(SYORINUM, RAINNUM, i) = sngOver - 簡易量(SYORINUM, RAINNUM, i)
End If
Else
'すべて貯留されたため簡易処理は行われない
簡易量(SYORINUM, RAINNUM, i) = 0
直放量(SYORINUM, RAINNUM, i) = 0
End If

'2012/04/01 水質基準種類追加
For w_Quality = 1 To 5
高級負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(高級量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "#,##0.0")
貯留負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(貯留流量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "#,##0.0")
簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(簡易量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "#,##0.0")
直放負荷量(SYORINUM, RAINNUM, i, w_Quality) = Format(直放量(SYORINUM, RAINNUM, i) * 水質(SYORINUM, RAINNUM, i, w_Quality), "#,##0.0")
Next w_Quality
End If

*****処理場放流負荷量*****
'2012/04/01 水質基準種類追加
For w_Quality = 1 To 5
高級放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(高級負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkoukyuuritsu(w_Quality) / 100),
"#,##0.0")
貯留放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(貯留負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkoukyuuritsu(w_Quality) / 100),
"#,##0.0")
簡易放流量(SYORINUM, RAINNUM, i, w_Quality) = Format(簡易負荷量(SYORINUM, RAINNUM, i, w_Quality) * (1 - sngkaniritsu(w_Quality) / 100), "#,##0.0")
直放放流量(SYORINUM, RAINNUM, i, w_Quality) = 直放負荷量(SYORINUM, RAINNUM, i, w_Quality)

```

Next w_Quality

***** 1 降雨分の集計*****

合計高級量(SYORINUM, RAINNUM) = 合計高級量(SYORINUM, RAINNUM) + 高級量(SYORINUM, RAINNUM, i)

合計貯留量(SYORINUM, RAINNUM) = 貯留量(SYORINUM, RAINNUM)

合計簡易量(SYORINUM, RAINNUM) = 合計簡易量(SYORINUM, RAINNUM) + 簡易量(SYORINUM, RAINNUM, i)

合計直放量(SYORINUM, RAINNUM) = 合計直放量(SYORINUM, RAINNUM) + 直放量(SYORINUM, RAINNUM, i)

'2012/04/01 水質基準種類追加

For w_Quality = 1 To 5

合計高級負荷量(SYORINUM, RAINNUM, w_Quality) = 合計高級負荷量(SYORINUM, RAINNUM, w_Quality) + 高級負荷量(SYORINUM, RAINNUM, i, w_Quality)

合計貯留負荷量(SYORINUM, RAINNUM, w_Quality) = 合計貯留負荷量(SYORINUM, RAINNUM, w_Quality) + 貯留負荷量(SYORINUM, RAINNUM, i, w_Quality)

合計簡易負荷量(SYORINUM, RAINNUM, w_Quality) = 合計簡易負荷量(SYORINUM, RAINNUM, w_Quality) + 簡易負荷量(SYORINUM, RAINNUM, i, w_Quality)

合計直放負荷量(SYORINUM, RAINNUM, w_Quality) = 合計直放負荷量(SYORINUM, RAINNUM, w_Quality) + 直放負荷量(SYORINUM, RAINNUM, i, w_Quality)

合計高級放流量(SYORINUM, RAINNUM, w_Quality) = 合計高級放流量(SYORINUM, RAINNUM, w_Quality) + 高級放流量(SYORINUM, RAINNUM, i, w_Quality)

合計貯留放流量(SYORINUM, RAINNUM, w_Quality) = 合計貯留放流量(SYORINUM, RAINNUM, w_Quality) + 貯留放流量(SYORINUM, RAINNUM, i, w_Quality)

合計簡易放流量(SYORINUM, RAINNUM, w_Quality) = 合計簡易放流量(SYORINUM, RAINNUM, w_Quality) + 簡易放流量(SYORINUM, RAINNUM, i, w_Quality)

合計直放放流量(SYORINUM, RAINNUM, w_Quality) = 合計直放放流量(SYORINUM, RAINNUM, w_Quality) + 直放放流量(SYORINUM, RAINNUM, i, w_Quality)

Next w_Quality

'計算結果をグローバル変数に格納

gsngKokyu(SYORINUM, RAINNUM, i) = 高級量(SYORINUM, RAINNUM, i)

gsngChoryuRyuryo(SYORINUM, RAINNUM, i) = 貯留流量(SYORINUM, RAINNUM, i)

gsngChoryu(SYORINUM, RAINNUM, i) = 貯留量(SYORINUM, RAINNUM)

gsngKani(SYORINUM, RAINNUM, i) = 簡易量(SYORINUM, RAINNUM, i)

gsngChokuhou(SYORINUM, RAINNUM, i) = 直放量(SYORINUM, RAINNUM, i)

'2012/04/01 水質基準種類追加

For w_Quality = 1 To 5

gsngSuishitu(SYORINUM, RAINNUM, i, w_Quality) = 水質(SYORINUM, RAINNUM, i, w_Quality)

gsngKokyu_Fuka(SYORINUM, RAINNUM, i, w_Quality) = 高級負荷量(SYORINUM, RAINNUM, i, w_Quality)

gsngChoryu_Fuka(SYORINUM, RAINNUM, i, w_Quality) = 貯留負荷量(SYORINUM, RAINNUM, i, w_Quality)

gsngKani_Fuka(SYORINUM, RAINNUM, i, w_Quality) = 簡易負荷量(SYORINUM, RAINNUM, i, w_Quality)

gsngChokuhou_Fuka(SYORINUM, RAINNUM, i, w_Quality) = 直放負荷量(SYORINUM, RAINNUM, i, w_Quality)

gsngKokyu_Houryu(SYORINUM, RAINNUM, i, w_Quality) = 高級放流量(SYORINUM, RAINNUM, i, w_Quality)

gsngChoryu_Houryu(SYORINUM, RAINNUM, i, w_Quality) = 貯留放流量(SYORINUM, RAINNUM, i, w_Quality)

gsngKani_Houryu(SYORINUM, RAINNUM, i, w_Quality) = 簡易放流量(SYORINUM, RAINNUM, i, w_Quality)

gsngChokuhou_Houryu(SYORINUM, RAINNUM, i, w_Quality) = 直放放流量(SYORINUM, RAINNUM, i, w_Quality)

Next w_Quality

```

Next i
If 合計直放量(SYORINUM, RAINNUM) > 0 Then
    放流回数(intRyuukiNo) = 放流回数(intRyuukiNo) + 1
End If
gintHouryuCount(SYORINUM) = 放流回数(intRyuukiNo)                '放流回数

***** 全降雨分の集計*****
年間合計高級量(intRyuukiNo) = 年間合計高級量(intRyuukiNo) + 合計高級量(SYORINUM, RAINNUM)
年間合計貯留量(intRyuukiNo) = 年間合計貯留量(intRyuukiNo) + 合計貯留量(SYORINUM, RAINNUM)
年間合計簡易量(intRyuukiNo) = 年間合計簡易量(intRyuukiNo) + 合計簡易量(SYORINUM, RAINNUM)
年間合計直放量(intRyuukiNo) = 年間合計直放量(intRyuukiNo) + 合計直放量(SYORINUM, RAINNUM)

For w_Quality = 1 To 5
    年間合計高級放流負荷量(intRyuukiNo, w_Quality) = 年間合計高級放流負荷量(intRyuukiNo, w_Quality) + 合計高級放流負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計貯留負荷量(intRyuukiNo, w_Quality) = 年間合計貯留負荷量(intRyuukiNo, w_Quality) + 合計貯留負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計簡易負荷量(intRyuukiNo, w_Quality) = 年間合計簡易負荷量(intRyuukiNo, w_Quality) + 合計簡易負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計直放負荷量(intRyuukiNo, w_Quality) = 年間合計直放負荷量(intRyuukiNo, w_Quality) + 合計直放負荷量(SYORINUM, RAINNUM, w_Quality)

    年間合計高級放流負荷量(intRyuukiNo, w_Quality) = 年間合計高級放流負荷量(intRyuukiNo, w_Quality) + 合計高級放流負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計貯留放流負荷量(intRyuukiNo, w_Quality) = 年間合計貯留放流負荷量(intRyuukiNo, w_Quality) + 合計貯留放流負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計簡易放流負荷量(intRyuukiNo, w_Quality) = 年間合計簡易放流負荷量(intRyuukiNo, w_Quality) + 合計簡易放流負荷量(SYORINUM, RAINNUM, w_Quality)
    年間合計直放放流負荷量(intRyuukiNo, w_Quality) = 年間合計直放放流負荷量(intRyuukiNo, w_Quality) + 合計直放放流負荷量(SYORINUM, RAINNUM, w_Quality)

Next w_Quality

'計算結果をグローバル変数に格納
gsngKokyu_Year(SYORINUM) = 年間合計高級量(intRyuukiNo)                '高級量
gsngChoryu_Year(SYORINUM) = 年間合計貯留量(intRyuukiNo)            '貯留量
gsngKani_Year(SYORINUM) = 年間合計簡易量(intRyuukiNo)            '簡易量
gsngChokuhou_Year(SYORINUM) = 年間合計直放量(intRyuukiNo)        '直放量

For w_Quality = 1 To 5
    gsngKokyu_Fuka_Year(SYORINUM, w_Quality) = 年間合計高級放流負荷量(intRyuukiNo, w_Quality)                '高級放流量
    gsngChoryu_Fuka_Year(SYORINUM, w_Quality) = 年間合計貯留放流負荷量(intRyuukiNo, w_Quality)                '貯留放流量
    gsngKani_Fuka_Year(SYORINUM, w_Quality) = 年間合計簡易放流負荷量(intRyuukiNo, w_Quality)                '簡易放流量
    gsngChokuhou_Fuka_Year(SYORINUM, w_Quality) = 年間合計直放放流負荷量(intRyuukiNo, w_Quality)                '直放放流量

    gsngKokyu_Houryu_Year(SYORINUM, w_Quality) = 年間合計高級放流負荷量(intRyuukiNo, w_Quality)                '高級放流量
    gsngChoryu_Houryu_Year(SYORINUM, w_Quality) = 年間合計貯留放流負荷量(intRyuukiNo, w_Quality)                '貯留放流量
    gsngKani_Houryu_Year(SYORINUM, w_Quality) = 年間合計簡易放流負荷量(intRyuukiNo, w_Quality)                '簡易放流量
    gsngChokuhou_Houryu_Year(SYORINUM, w_Quality) = 年間合計直放放流負荷量(intRyuukiNo, w_Quality)                '直放放流量

```

```

Next w_Quality
GoTo 17005

subSyorijyouCalc_Exit:
Exit Sub
subSyorijyouCalc_Err:
MsgBox Err.Description
Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation] : " & "subSyorijyouCalc", "処理場計算にてエラーが発生しました。")
Resume subSyorijyouCalc_Exit

End Sub

*****
' 雨水吐き・ポンプ場計算
'
'
*****
Public Sub subUsuiCalc()
On Error GoTo subUsuiCalc_Err
Dim i As Integer
Dim n As Integer
Dim RAINNUM As Integer
Dim intKouuCount As Integer
Dim intStepSecond As Integer
Dim w_Quality As Integer

intKouuCount = UBound(gttypKouuData)
intStepSecond = CInt(gstrStepMinute) * 60 '1 ステップあたりの秒数を設定

ReDim QAAATres(RYUSU, intKouuCount) As Single
ReDim EfluTTres(RYUSU, intKouuCount) As Single
ReDim QAAATres(RYUSU) As Single
ReDim EfluTTres(RYUSU) As Single
ReDim ChosuiTTres(RYUSU) As Single
ReDim gintUsuiHouryuCount(RYUSU) As Integer

ReDim TFUKATres(RYUSU, intKouuCount, 5) As Single
ReDim EloadTTres(RYUSU, intKouuCount, 5) As Single
ReDim ChoTTres(RYUSU, intKouuCount, 5) As Single
ReDim TFUKATres(RYUSU, 5) As Single
ReDim EloadTTres(RYUSU, 5) As Single

```

```

ReDim ChoTTTres(RYUSU, 5) As Single
'すべての降雨で計算
For RAINNUM = 1 To intKouuCount
    MenuFblCalcStatus.Caption = "雨水吐きなど計算実行中です" & vbCrLf & "(" & RAINNUM & "/" & UBound(gttypKouuData) & "件目)"
    DoEvents
    For n = 1 To RYUSU
        If Usui(n) = 1 Then
            '雨水吐き
            GoTo 17010
        ElseIf Usui(n) = 2 Then
            'ポンプ場
            GoTo 17020
        End If
        '対策施設の計算処理(ハイパス時)
        If IsNumeric(gttypShisetsuPram(n).strNo) = True Then
            '対策施設
            GoTo 17040
        End If
    Next n
subUsuiCalc_Return:
    Next n
Next RAINNUM
'すべての計算が終わったら終了
GoTo subUsuiCalc_Exit
17010
'*****雨水吐き計算*****
17011
'***** 1降雨分の集計*****
For i = 1 To gttypKouuData(RAINNUM).lngStepCount
    '1ステップ分の総流量を計算して加算します。
    QAAATres(n, RAINNUM) = QAAATres(n, RAINNUM) + (QAA(RAINNUM, n, i) * intStepSecond)
    EfluTres(n, RAINNUM) = EfluTres(n, RAINNUM) + (Eflu(RAINNUM, n, i) * intStepSecond)
Next i
'***** 全降雨分の集計 (流量) *****
QAAATres(m) = QAAATres(n) + QAAATres(n, RAINNUM)

```

```

EfluTTres(n) = EfluTTres(n) + EfluTTres(n, RAINNUM)
ChosuiTres(n) = ChosuiTres(n) + Chosui(RAINNUM, n, gtypKouuData(RAINNUM).lngStepCount) '最後のステップで貯留されている量を加算

If EfluTTres(n, RAINNUM) > 0 Then
    gintUsuiHouryuCount(n) = gintUsuiHouryuCount(n) + 1
End If

17014
***** 1 降雨分の集計*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    For w_Quality = 1 To 5
        '1ステップ分の総流量を計算して加算します。
        TFUKATres(n, RAINNUM, w_Quality) = TFUKATres(n, RAINNUM, w_Quality) + (TFUKA(RAINNUM, w_Quality, n, i) * intStepSecond)
        EloadTTres(n, RAINNUM, w_Quality) = EloadTTres(n, RAINNUM, w_Quality) + (Eload(RAINNUM, w_Quality, n, i) * intStepSecond)
        ChoTTres(n, RAINNUM, w_Quality) = ChoTTres(n, RAINNUM, w_Quality) + (ChoT(RAINNUM, w_Quality, n, i) * intStepSecond)
    Next w_Quality
Next i

***** 全降雨分の集計*****
For w_Quality = 1 To 5
    TFUKATres(n, w_Quality) = TFUKATres(n, w_Quality) + TFUKATres(n, RAINNUM, w_Quality)
    EloadTTres(n, w_Quality) = EloadTTres(n, w_Quality) + EloadTTres(n, RAINNUM, w_Quality)
    ChoTTres(n, w_Quality) = ChoTTres(n, w_Quality) + ChoTTres(n, RAINNUM, w_Quality)
Next w_Quality

17015
GoTo subUsuiCalc_Return

17020
***** ポンプ場等計算*****
17021
***** 1 降雨分の集計*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    '1 ステップ分の総流量を計算して加算します。
    QAAATres(n, RAINNUM) = QAAATres(n, RAINNUM) + (QAA(RAINNUM, n, i) * intStepSecond)
    EfluTres(n, RAINNUM) = EfluTres(n, RAINNUM) + (Eflu(RAINNUM, n, i) * intStepSecond)
Next i

***** 全降雨分の集計 (流量) *****
QAAATres(n) = QAAATres(n) + QAAATres(n, RAINNUM)

```



```

EfluTTres(n) = EfluTTres(n) + EfluTTres(n, RAINNUM)
ChosuiTres(n) = ChosuiTres(n) + Chosui(RAINNUM, n, gtypKouuData(RAINNUM).lngStepCount)
If EfluTres(n, RAINNUM) > 0 Then
    gintUsuiHouryuCount(n) = gintUsuiHouryuCount(n) + 1
End If

17023 ***** 負荷量計算結果*****
17024 ***** 1 降雨分の集計*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    For w_Quality = 1 To 5
        '1 ステップ分の総流量を計算して加算します。
        TFUKATres(n, RAINNUM, w_Quality) = TFUKATres(n, RAINNUM, w_Quality) + (TFUKA(RAINNUM, w_Quality, n, i) * intStepSecond)
        EloadTres(n, RAINNUM, w_Quality) = EloadTres(n, RAINNUM, w_Quality) + (Eload(RAINNUM, w_Quality, n, i) * intStepSecond)
        ChoTTres(n, RAINNUM, w_Quality) = ChoTTres(n, RAINNUM, w_Quality) + (ChoT(RAINNUM, w_Quality, n, i) * intStepSecond)
    Next w_Quality
Next i

***** 全降雨分の集計*****
For w_Quality = 1 To 5
    TFUKATres(n, w_Quality) = TFUKATres(n, w_Quality) + TFUKATres(n, RAINNUM, w_Quality)
    EloadTres(n, w_Quality) = EloadTres(n, w_Quality) + EloadTres(n, RAINNUM, w_Quality)
    ChoTTres(n, w_Quality) = ChoTTres(n, w_Quality) + ChoTTres(n, RAINNUM, w_Quality)
Next w_Quality

17025 GoTo subUsuiCalc_Return

17040 ***** 対策施設等計算*****
17041 ***** 1 降雨分の集計*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    QAAATres(n, RAINNUM) = QAAATres(n, RAINNUM) + (QAA(RAINNUM, n, i) * intStepSecond)
    EfluTres(n, RAINNUM) = EfluTres(n, RAINNUM) + (Eflu(RAINNUM, n, i) * intStepSecond)
Next i

***** 全降雨分の集計 (流量) *****
QAAATres(n) = QAAATres(n) + QAAATres(n, RAINNUM)
EfluTres(n) = EfluTres(n) + EfluTres(n, RAINNUM)
ChosuiTres(n) = ChosuiTres(n) + Chosui(RAINNUM, n, gtypKouuData(RAINNUM).lngStepCount)

```

```

If EfluTres(n, RAINNUM) > 0 Then
    gintUsuiHouryuCount(n) = gintUsuiHouryuCount(n) + 1
End If

***** 負荷量計算結果*****
***** 1 降雨分の集計*****
For i = 1 To gtypKouuData(RAINNUM).lngStepCount
    For w_Quality = 1 To 5
        TFUKATres(n, RAINNUM, w_Quality) = TFUKATres(n, RAINNUM, w_Quality) + (TFUKA(RAINNUM, w_Quality, n, i) * intStepSecond)
        EloadTres(n, RAINNUM, w_Quality) = EloadTres(n, RAINNUM, w_Quality) + (Eload(RAINNUM, w_Quality, n, i) * intStepSecond)
        ChoTTres(n, RAINNUM, w_Quality) = ChoTTres(n, RAINNUM, w_Quality) + (ChoT(RAINNUM, w_Quality, n, i) * intStepSecond)
    Next w_Quality
Next i

***** 全降雨分の集計*****
For w_Quality = 1 To 5
    TFUKATres(n, w_Quality) = TFUKATres(n, w_Quality) + TFUKATres(n, RAINNUM, w_Quality)
    EloadTres(n, w_Quality) = EloadTres(n, w_Quality) + EloadTres(n, RAINNUM, w_Quality)
    ChoTTres(n, w_Quality) = ChoTTres(n, w_Quality) + ChoTTres(n, RAINNUM, w_Quality)
Next w_Quality

If EfluTres(n, RAINNUM) > 0 Then
    gintUsuiHouryuCount(n) = gintUsuiHouryuCount(n) + 1
End If

GoTo subUsuiCalc_Return

subUsuiCalc_Exit:
Exit Sub
subUsuiCalc_Err:
MsgBox Err.Description
Call OutputLog(g_LogFileName, g_LOGMODE_ERROR, "[Mod_Calculation] : " & "subUsuiCalc", "雨水吐き・ポンプ場計算にてエラーが発生しました。")
Resume subUsuiCalc_Exit

End Sub

```

参考文献

- 1) 合流式下水道改善対策指針と解説—2002年版—, (社)下水道協会
- 2) 流出解析モデル利活用マニュアル, 1999年3月, (財)下水道新技術推進機構
- 3) 合流式下水道改善計画策定のためのモニタリングマニュアル(案)
2003年3月, (財)下水道新技術推進機構
- 4) 合流式下水道越流水対策と暫定指針, 1982年版, 日本下水道協会
- 5) 下水道を考慮した都市域における氾濫シミュレーション技術(PWR Iモデル)の開発, 土木学会第57回年次学術講演会資料, 平成14年9月
- 6) 下水道管路施設設計の合理化に関する調査報告書(IV)—合流式下水道の改良に関する調査—, 土木研究資料第1704号, 昭和56年8月, 建設省土木研究所下水道研究室
- 7) 合流式下水道改善策に関する調査, 昭和61年3月, 東京都下水道局
- 8) 都市雨水による汚濁負荷の流出特性と水質汚濁負荷に関する研究, 2000年2月, 中村栄一
- 9) 下水道雨水浸透技術マニュアル—2001年6月—, (財)下水道新技術推進機構
- 10) JAMSOOP 操作説明書, 平成18年4月, 国土交通省 国土技術政策総合研究所